

Blender GameEngine a jeho využití

Blender GameEngine and its application

Michael Žídek



Univerzita Tomáše Bati ve Zlíně
Fakulta aplikované informatiky
akademický rok: 2011/2012

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení: **Michael ŽÍDEK**
Osobní číslo: **A09320**
Studijní program: **B 3902 Inženýrská informatika**
Studijní obor: **Informační a řídicí technologie**

Téma práce: **Blender Game Engine a jeho využití**

Zásady pro vypracování:

1. Seznamte se podrobně s programem Blender, zaměřte pozornost zejména na část pro tvorbu interaktivních aplikací.
2. Popište základní prostředí programu Blender, včetně nástrojů pro tvorbu interaktivních aplikací.
3. Navrhněte a vytvořte názornou interaktivní aplikaci v prostředí Blender, využívající implementovaný GAMEENGINE.
4. Vytvořte podrobnou dokumentaci k vytvořené aplikaci.
5. Vytvořte podrobnou elektronickou příručku k tvorbě interaktivních aplikací pro aktuální verzi programu Blender.

Rozsah bakalářské práce:

Rozsah příloh:

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

Seznam odborné literatury:

1. POKORNÝ PAVEL. Blender – naučte se 3D grafiku. 2.vyddání, Praha BEN – technická literatura, 2009. ISBN 80-7300-244-2.
2. Grafika online. Animace v Blenderu. Blender [Online].2011.Dostupné z WWW: <http://www.grafika.cz/art/3d/clanek2047803217.html>
3. Blender [Online].2011.dostupné z WWW: <http://www.blender.org>.
4. Doc:Manual [Online].2011.BlenderWiki.Dostupné z WWW: <http://wiki.blender.org/index.php/DocManual>.
5. Blender Artists Komunity. Diskuzní fórum [Online].2011.dostupné z WWW: <http://www.blenderartists.org>

Vedoucí bakalářské práce:

Ing. Pavel Pokorný, Ph.D.

Ústav počítačových a komunikačních systémů

Datum zadání bakalářské práce:

24. února 2012

Termín odevzdání bakalářské práce:

8. června 2012

Ve Zlíně dne 24. února 2012

prof. Ing. Vladimír Vašek, CSc.
děkan



prof. Ing. Vladimír Vašek, CSc.
ředitel ústavu

ABSTRAKT

Tato bakalářská práce se zaměřila na tvorbu 3D interaktivních aplikací a her ve vývojovém programu Blender. K tomuto programu je vypracována literární rešerše, ve které je uveden popis programu, historický vývoj a vlastnosti programu. V teoretické části je popsáno základní prostředí pro tvorbu 3D scény. V praktické části je pozornost zaměřena především na tvorbu 3D interaktivních aplikací, dostupných nástrojů pro tvorbu scény, prostředí pro tvorbu herní logiky, práci s GameBlender engine, a tvorbou menu nabídky.

Cílem praktické části bakalářské práce je vytvoření podrobné příručky pro tvorbu 3D interaktivních aplikací, která by mohla sloužit jako pomocný studijní materiál, a také vytvořit názornou aplikaci, ve které budou demonstrovány schopnosti a možnosti programu Blender.

Klíčová slova: Blender, 3D grafika, GameBlender, herní engine, tvorba 3D her a interaktivních aplikací

ABSTRACT

This Bachelor thesis focuses on the creating interactive 3D applications and games in program Blender. The history, development and features of this program were obtained in the bibliographic search. The theoretical part discusses the basic environment for creating 3D scenes. The practical part of the thesis concentrates on the creating interactive 3D applications and on tools available for creating of scenes. Moreover it discusses the environment for game logic, work with GameBlender engine, and the creation of menu offerings.

The practical part has two main goals. The first goal is to create a detailed guide for creating interactive 3D applications, which would serve as an additional study textbook. The second goal is to create a visual application on which all the possibilities of program Blender will be demonstrated.

Keywords: Blender, 3D graphics, GameBlender, game engine, creating interactive 3D games and applications

Chtěl bych zde poděkovat svému vedoucímu, panu Ing. Pavlu Pokornému, Ph.D, za odborné vedení a rady při tvorbě mé bakalářské práce. Dále bych rád poděkoval mým spolužákům, kamarádům a všem lidem, kteří mi přispěli dobrou radou k vyřešení problémů spojených s tvorbou 3D aplikace.

Také bych chtěl poděkovat mé rodině, především rodičům za to, že jsem mohl studovat na Fakultě Aplikované Informatiky na Univerzitě Tomáše Bati ve Zlíně, že mně celé roky podporovali a věřili ve mně.

Motto:

„Strach je jako houpací křeslo... Houpáš se, ale nikam tě to neposune.“

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.
- že odevzdaná verze bakalářské práce a verze elektronická nahraná do IS/STAG jsou totožné.

Ve Zlíně

.....
podpis diplomanta

OBSAH

ÚVOD.....	9
I TEORETICKÁ ČÁST.....	10
1 PROGRAM BLENDER.....	11
1.1 CHARAKTERISTIKA PROGRAMU	11
1.1.1 Hlavní rysy	12
1.2 HISTORIE BLENDERU.....	13
1.2.1 Verze Blenderu a významná data.....	14
1.3 VERZE BLENDERU S GAMEENGINE	14
1.3.1 Blender 2.33 (Duben 2004).....	14
1.3.2 Blender 2.34 (Červenec 2004)	15
1.3.3 Blender 2.35a (Listopad 2004).....	16
1.3.4 Blender 2.36 (Prosinec 2004).....	16
1.3.5 Blender 2.37(Červen 2005).....	17
1.3.6 Blender 2.40 (prosinec 2005).....	17
1.3.7 Blender 2.41 (Leden 2006)	18
1.3.8 Blender 2.44 (Květen 2007).....	19
1.3.9 Blender 2.46 (Květen 2008).....	19
1.3.10 Blender 2.50 (Březen 2010)	20
1.3.11 Blender 2.60 (Říjen 2011).....	21
1.3.12 Blender 2.61 (Prosinec 2011).....	21
1.3.13 Blender 2.62 (Březen 2012)	22
1.3.14 Blender 2.63 (Duben 2012).....	23
2 PROSTŘEDÍ PRO TVORBU 3D INTERAKTIVNÍCH APLIKACÍ.....	24
2.1 ZÁKLADNÍ UŽIVATELSKÉ ROZHRAŇÍ PROGRAMU BLENDER.....	24
2.2 GAME LOGIC PANEL	26
2.2.1 Senzory.....	27
2.2.2 Kontroléry	27
2.2.3 Akční členy	27
2.3 UV / IMAGE EDITOR	28
II PRAKTICKÁ ČÁST	29
3 DOKUMENTACE K VYTVOŘENÉ INTERAKTIVNÍ APLIKACI.....	30
3.1 VYTVÁŘENÍ OBJEKTŮ VE SCÉNĚ	30
3.2 NASTAVENÍ MATERIÁLŮ.....	31
3.2.1 Properties.....	31
3.2.1.1 Material	32
3.2.1.2 Textures	33
3.2.1.3 UV / Image Editor.....	34
3.2.2 Postup tvorby 3D obsahu	35
3.2.3 Vytváření herní logiky a kolizí	40
3.2.3.1 Objekt Player	40
3.2.3.2 Menu	43
ZÁVĚR	47
CONCLUSION	48

SEZNAM POUŽITÉ LITERATURY.....	49
SEZNAM POUŽITÝCH SYMBOLŮ A ZKRATEK.....	50
SEZNAM OBRÁZKŮ	51
SEZNAM PŘÍLOH.....	53

ÚVOD

Tato bakalářská práce se zabývá 3D interaktivními aplikacemi, jejich tvorbou hlavně v grafickém programu Blender, zejména pak v jeho implementované části GameBlender. Tato bakalářská práce je psána pro aktuální verzi programu Blender (2.61) v době, kdy bylo oficiálně předáno zadání této práce.

Blender obsahuje mnoho nástrojů pro tvorbu 3D grafického obsahu, modelování scény, renderování obrázků a animací, a hlavně pro tvorbu a přehrávání interaktivních aplikací. Přednosti programu Blender jsou zejména ty, že je hardwarově nenáročný, a jeho prostředí je uživatelsky přijatelnější z hlediska rychlé a atraktivní tvorby aplikací jak pro začátečníky, tak i pro pokročilé tvůrce 3D her. Další výhodou je, že Blender je multiplatformní a díky malé velikosti programu (nejnovější verze Blender 2.61 má 38MB) je vhodný i pro přenosovou verzi. Jedná se o Open source software s volně šiřitelným zdrojovým kódem pod licencí GNU GPL.

Tato Bakalářská práce má za úkol pokusit se přiblížit část programu Blender, která se nazývá GameBlender. Jedná se o implementovanou část real-time 3D enginu, která má široké využití, ať už u tvorby interaktivních aplikací, přes reklamy s možností začlenění do internetových prohlížečů, až k tvorbě her.

GameBlender se specializuje na tvorbu 3D interaktivních aplikací, her a simulací chování objektů 3D obsahu scény podle zákonů fyziky. Implementovaná část programu Blender, známá jako Game Engine, zajišťuje zpracování vstupních signálů ze zdrojů, kterým jsou přiřazené logické bloky, dále zajišťuje výpočet kolizního systému a fyzikálních zákonů, jimiž se objekty budou v simulaci vytvořené reality chovat. Tyto signály se dále posílají na výstupní zařízení, které provádí požadovanou akci.

Součástí této bakalářské práce je i elektronická příručka pro tvorbu 3D interaktivních aplikací, ve které jsou podrobněji popsány nástroje a funkce důležité pro tvorbu interaktivních aplikací a her, zejména pro tvorbu herní logiky a mapování textur na objekty pomocí UV / Image editoru.

Dále na přiloženém CD, které je také součástí bakalářské práce, je přiložena názorná ukázka interaktivní aplikace, ve které jsou použity funkce a nástroje, které GameEngine nabízí a jsou také popsány v praktické části této bakalářské práci, která je soustředěna na popis tvorby této aplikace.

I. TEORETICKÁ ČÁST

1 PROGRAM BLENDER

1.1 Charakteristika programu

Blender je multiplatformní open source aplikace zaměřená na vytváření 3D modelů, animací, rendering, postprodukční činnost a v neposlední řadě na tvorbu a přehrání interaktivních aplikací. Multiplatformní znamená, že je dostupný pro mnoho operačních systémů, jako jsou Windows, Linux, Mac OS, a další. To vše proto, že je založen na OpenGL grafické knihovně.[1]

Jeho hlavní výhodou je velikost programu, která je malá (u současné verze Blender 2.61 má velikost instalačního souboru 27 MB a složka .zip má pouhých 38 MB). Mezi další výhody patří hardwarová nenáročnost. Výjimkou zde je grafická karta, na níž závisí rychlost a výkonnost programu (výpočty zobrazovaného 3D obsahu), nicméně Blender jako takový funguje i na starších typech grafických karet.

Velkou předností je jeho uživatelské prostředí, v němž jsou veškeré položky umístěny přehledně a intuitivně. Blender jako program je vhodný jak pro úplné začátečníky, tak i pro lidi, kteří již s tvorbou interaktivního obsahu mají nějaké zkušenosti a přechází z jiných 3D grafických editorů. Výhodou také v neposlední řadě je velká komunita lidí, kteří pracují s programem Blender a radí si na svých fórech na internetu. Takovou malou nevýhodou je, že práce v něm je ze začátku pomalá, což se časem eliminuje, protože většina úkonů, potřebných pro tvorbu 3D obsahu, má své klávesové zkratky, po jejichž navyknutí se práce s tímto programem stává velmi rychlou a efektivní.

Blender získal začleněním herního engine možnost tvorby 3D interaktivního obsahu a jeho zobrazování a přehrávání v internetových prohlížečích. Tato část se nazývá GameBlender a nabízí grafický editor pro tvorbu 3D scény bez nutnosti znalostí programování. “GameEngine“ má v sobě zabudované simulace kolizí, fyzikální prvky a jako celek zajišťuje interaktivitu.

Dá se prohlásit, že GameEngine slouží k tomu, aby k námi vytvořené virtuální realitě, která se skládá z objektů, které mohou představovat různé věci (budovy, lidi), bylo přiřazeno i chování (Fyzika, herní logika). Herní objekty v této realitě, neboli “GameObjects“ mají své vlastní sady nástrojů, jako jsou logické bloky (LogicBricks) a vlastnosti (Properties). Logickými bloky jsou v tomto případě myšleny senzory, akční členy a kontrolory.

1.1.1 Hlavní rysy

- malá velikost programu a HW nenáročnost
- Blender je multiplatformní a zcela zdarma
- velká komunita lidí pracujících s programem Blender
- mnoho informací a návodů na internetu
- díky intuitivně zpracovanému vývojovému prostředí je vhodný na tvorbu 3D obsahu a her jak pro pokročilé, tak i pro úplné začátečníky
- plně přizpůsobitelné rozhraní
- zpětná podpora pro všechny úrovně
- stejné uživatelské rozhraní pro všechny platformy
- přístup k Python skriptům pro vlastní řešení
- velké množství 3D objektů a polygonových sítí
- modelování pomocí různých metod vybírání ploch, hran a bodů
- rychlá práce s pomocí pohybových křivek
- rychlé přepínání mezi jednotlivými snímky animací díky časové ose
- přehrávání zvuku, míchání a podpora pro synchronizaci zvuku
- přístup k Python skriptům pro vlastní animační efekty
- rendering hran pro vykreslování stínů
- interaktivní panel pro náhled vizualizace objektů v okně 3D View
- jednoduchá tvorba herní logiky ve speciálním panelu bez nutnosti znalosti programovacího jazyku
- Python API pro používání skriptů
- podpora všech OpenGL režimů osvětlení
- plně definována pokročilá herní logika
- možnost exportování vytvořených aplikací přímo do spustitelných souborů s příponou .EXE

1.2 Historie Blenderu

V roce 1988 bylo Tonem Roosendaalem v Holandsku založeno studio NeoGeo, které se brzy po svém vzniku stalo jedním z předních animačních studií v Evropě. Toto studio vytvářelo významné oceněné produkty (European Corporate Video Awards v roce 1993 a 1995) pro mnoho korporací, zejména pro Philips. Ton Roosendaal byl v zodpovědný mimo jiné za výtvarné směry a vývoj softwaru. V roce 1995 bylo usouzeno, že aktuální sada 3D nástrojů je zastaralá a je nutno ji od základu přepsat. Práce na tomto projektu započaly v tentýž rok, a za nedlouho byl na světě program, který je dnes znám jako Blender. V době, kdy se dokončovaly práce na tomto projektu, přišla i myšlenka, že by Blender mohli používat i jiní umělci, nejen zaměstnanci firmy NeoGeo.

Roku 1998 byla založena firma NaN (Not a Number), se zaměřením na širší trh a rozvoj programu Blender. Ideou NaN bylo vytvoření a distribuce zdarma malého, multiplatformního programu pro tvorbu 3D grafiky. NaN doufala, že vytvoří profesionální nástroj pro modelování a animaci, který by byl dostupný komukoliv. NaN sázela na služby kolem Blenderu a v roce 1999 se firma poprvé zúčastnila konference Siggraph se snahou o větší propagaci programu Blender. Blender zde měl velký ohlas a zájem o něj se projevil nejen ze strany novinářů, ale i ze strany návštěvníků konference, což potvrdilo jeho potenciál.

Na základě této konference získala firma od investorů kapitál, který umožnil rychlý růst firmy a pracovníků, kteří pracovali na zlepšení a podpoře Blenderu. V létě roku 2000 byl vydán Blender 2.0, ve kterém bylo přidáno integrované herní rozhraní k 3D nástrojům (GameEngine).

V roce 2001 byl vydán první komerční produkt – Blender Publisher, nicméně malým prodejem a tíživou finanční situací firma zkrachovala. Roku 2002 byla Tonem Roosendaalem založena nekomerční společnost Blender Foundation, která měla za cíl najít způsob jak pokračovat v rozvoji a podpoře Blenderu jako Open Source projektu. V Říjnu 2002 byl Blender vydán pod licencí GNU General Public Licence (GPL). Do dnes trvá rozvoj Blenderu na kterém pracuje kolektiv mnoha dobrovolníků z celého světa, které vede sám Ton Roosendaal. [3]

1.2.1 Verze Blenderu a významná data

- 1.00 – Leden 1995, Blender vyvíjený v animačním studiu NeoGeo
- 1.23 – Leden 1998, SGI verze publikována na webu, IrisGL
- 1.30 – Duben 1998, Verze pro Linux a FreeBSD, výstup do OpenGL a X
- 1.3X – Červen 1998, Založena NaN
- 1.4X – Září 1998, Uvolněna alfa verze pro Sun a Linux
- 1.50 – Listopad 1998, Publikován první manuál
- 1.60 – Duben 1999, C-key (nástroj pro uzamčení, 95\$), uvolněna verze pro Windows
- 1.6X – Červen 1999, Uvolněna verze pro BeOS a PPC
- 1.80 – Červen 2000, Ukončení C-key, Blender opět Freeware
- 2.00 – Srpen 2000, Interaktivní 3D a real-time engine
- 2.10 – Prosinec 2000, Nový Engine, Fyzika a Python
- 2.20 – Srpen 2001, Character animation system
- 02.21 – Říjen 2001, Vypuštěn Blender Publisher
- 2.2X – Prosinec 2001, Mac OSX verze
- 13. Října 2002, Blender přechází na Open Source, první Blender Konference

1.3 Verze Blenderu s GameEngine

Tato kapitola se zaměřila pouze na vybrané verze programu Blender se zaměřením především na novinky v jeho implementované části GameEngine oproti předešlým verzím.

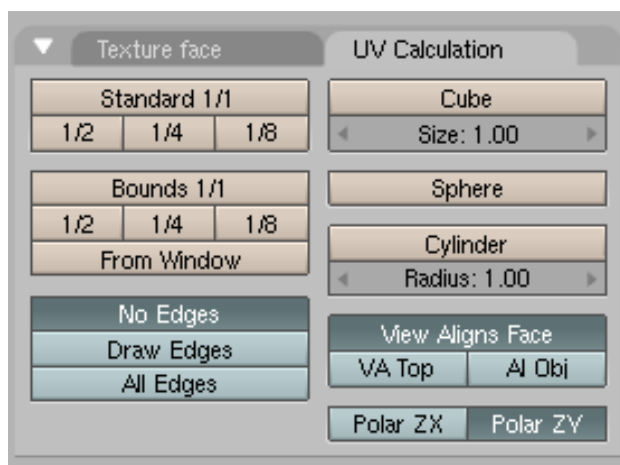
1.3.1 Blender 2.33 (Duben 2004)

Původní herní engine v Blenderu vydaný ve verzi 2.10 v počátku roku 2001, byl napsán Erwinem Coumansem (hlavní architektura) a Ginem van den Bergem (detekce kolizí a fyziky). Byl to významný upgrade starého enginu, který byl vydán s verzí 2.0 v Červnu 2000.

Hlavním cílem bylo poskytnout 3D umělcům přístup k vytváření interaktivního 3D obsahu. Komerční verze Blenderu podporovala samotné hráče. Multiplatformní webový plug-in, a možnosti jak podepsat a uzamknout obsah.

V této verzi byl připojen GameEngine s opravenou fyzikou na úroveň 2.25. cílem bylo, aby vytvořená verze byla plně kompatibilní s předešlou verzí programu Blender. Verze 2.33 ovšem překvapila v tom, že plnohodnotně nahradila starou verzi. Největší prací bylo, že je zde vytvořena nová knihovna pro detekci kolizí (Solid library 3.5), která však nebyla kompatibilní s verzí 2.5, proto se mohly objevit některé drobné chyby s kolizí a výkonem (zejména s rychlostí).

Další úpravou je nové funkce pro přidělování a výpočet UV souřadnic pro mapování textur v UV Editoru. V UV panelu přibyla karta pro UV kalkulace „UV Calculation“ (Obr. 1), která se zobrazuje v editačním panelu. Na této kartě byla veškerá nastavení potřebná pro práci s UV mapou. V UV editoru také přibyla nová volba výběru, a to „Aktive Face select“, která slouží k výběru ploch, které mají být mapovány. Výběr se provádí pomocí pravého tlačítka myši (RMB).[6]



Obr. 1: Karta UV kalkulace

1.3.2 Blender 2.34 (Červenec 2004)

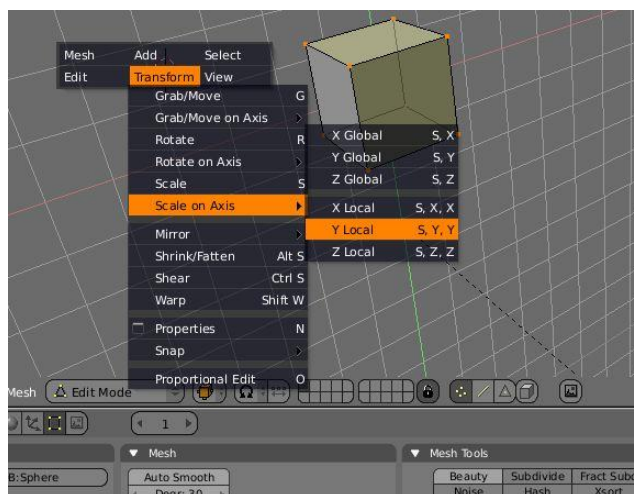
U herního enginu došlo k mnoha změnám a opravám chyb. Znovu je možné také ukládat jako samostatně běžící soubory díky Runtime. V systémech Windows je možné tyto soubory exportovat jako spustitelné soubory s příponou .EXE, na OSX jako Aplikace s příponou .app. Dále se zde může nastavovat režim celé obrazovky (Full Screen), kde se mohou nastavit různá rozlišení, ale je dobré se držet společných dimenzovaných rozlišení (640 * 480, 800 * 600, 1024 * 768 atd.).

Jako další novinkou je Stereo režim, který se používá pro zobrazení životního prostředí prostřednictvím duálního zobrazovacího systému, jako je pár polarizovaných projektorů nebo 3D brýle. V Runtime se může nově také nastavit bitová hloubka palety barev pro soubor, kde nižší bitová hloubka umožňovala zdroji běžet rychleji.[6]

1.3.3 Blender 2.35a (Listopad 2004)

Byl upraven vnitřní systém času a byly vylepšeny kolize, což vedlo k zrychlení fyzikálních výpočtů. Od této verze je odstraněno dvojí testování kolizí proti proniknutí. Logika a fyzika již není závislá na snímkové frekvenci (Frame Rate).[6] Logika se odehrává při snímkové frekvenci 30Hz a fyzika na 60Hz, což může být nastaveno jako výchozí pouze pomocí Pythonu. Některé logické bloky (např. IPO, Action, camera, track to Actuators) stále běží na frekvenci snímků pro plynulý pohyb.

Všechny verze nyní obsahují samostatný přehrávač.[5] Mimo jiné v této verzi také bylo zavedeno průhledné menu, které umožňuje v ústředním editoru definovat alfa hodnoty pro menu pozadí a předmětů a nechat si je ukazovat transparentní v uživatelském rozhraní.



Obr. 2: Průhledné menu[6]

1.3.4 Blender 2.36 (Prosinec 2004)

V této verzi přišlo mnoho vylepšení, a to jak v implementované části GameEngine, tak i v texturách. Objekty, které nemají přidělenou obrázkovou texturu, mohou být zobrazovány jako barevné tím, že jim je přidán materiál s nastavenou barvou. Ve 3D okně je nyní také možné nechávat si zobrazit přímo plochy a objekty s přiřazenými texturami.

Dalším vylepšením je, že jakýkoliv spustitelný soubor (s příponou .EXE), který je vyexportován z programu Blender, může být použit ve Windows jako spořič obrazovky.

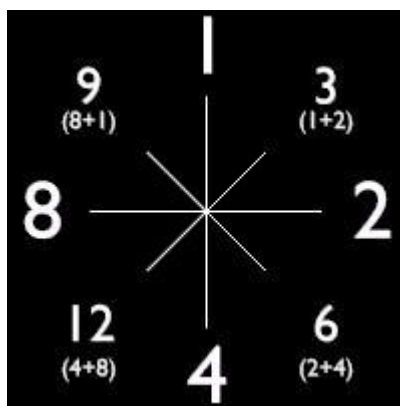
K tomu je ovšem nutné, aby byl pojmenovaný jako .SCR a pro jeho zobrazování musí být před použitím nainstalován. Toho se docílí kliknutím pravého tlačítka myši na soubor a zvolením položky instalovat.

Byl vyřešen i problém s OpenAL, který generoval signál, jež způsoboval, že po zastavení audia se program Blender vypnul. Fyzikální engine byl nastaven standardně na „Sumo“ kvůli čtení starších souborů.

1.3.5 Blender 2.37(Červen 2005)

V této verzi je snad největším vylepšením to, že je zde nově podpora senzoru pro Joystick. To má za následek, že tento senzor přidal podporu pro 4 osy, 18 tlačítek a 2 směrové kloboučky. Logické bloky jsou digitální, proto je možné k analogovému joysticku přistupovat pomocí Pythonu, který vrátí seznam 4 proměnných.

Čtení dat senzory z pohybových kloboučků joysticku se provádí za pomoci maskování řídicí hodnoty a to tak, že pro hodnoty na diagonálách, je potřeba přidat komponent hodnoty, tj. hodnota 9 se získá tím, že se sečtou hodnoty, které ji obklopují (Obr. 3)



Obr. 3: Hodnoty pro Joystick klobouky [6]

1.3.6 Blender 2.40 (prosinec 2005)

Tato verze přišla s podporou pro fyzicky založené animace, zabezpečuje lepší dynamiku tělesa a dají se zde zapékat vlastnosti pomocí GameEngine. Byly zde přidány také nové možnosti práce s IPO křivkami, kde každá IPO křivka nově může získat ovladač („Driver“) účelově, což ve své podstatě nahrazuje čas pro vstup pro konkrétní animovaný kanál s libovolnou hodnotou z jiného objektu.[6]

Blender GameEngine má novou fyzikální část, která se nazývá „Bullet“, která slouží k simulaci kolize a pohybu těles s tzv. „Rigid Body“ (tuhé tělo), a zároveň umožňuje zapékání IPO křivky, pro opakované použití v animovaných projektech.

Každý objekt, u kterého požadujeme, aby se choval dle fyzikálních zákonů, je potřeba aby mu byla v okně Logic, stisknuto tlačítko „Actor“ (Herec). (Obr. 4) Výběrem fyzikálního enginu Bullet můžeme také nastavovat gravitační konstantu, se kterou se poté bude ve fyzikálním enginu zjišťovat, jak rychle objekty padají na zem.



Obr. 4: Nastavení objektu jako Herec[5]

Další novinkou jsou nové volby možností pro zobrazení Statických částic (například vlasy), které do této doby mely obrázky zarovnaný na 1px široký „pramen“ a byly vykreslovány ve 3D pravidelné geometrii. Nově se dá použít anizotropního filtrování možností tak, aby bylo dosaženo realistického vzhledu vlasů.

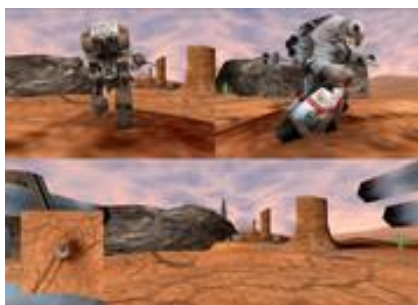
Dalšími nástroji zde byly možnost dělit plochy i celé objekty na menší části pomocí „Subdivide“ (dělení), booleovská operace „Intersection“ (průnik) a byl přidán nový nástroj „Rip“ (odtržení).[2]

1.3.7 Blender 2.41 (Leden 2006)

Tato verze vyšla přibližně po měsíci od předchozí verze 2.40. Ve verzi 2.41, která přinesla opravy chyb, došlo také k vylepšení ze jména v části GameEnginu, ve kterém jsou přidány řady funkcí, jakožto například GLSL shadery, UV mapování a návrat „Armatura“ systému.

V předešlých verzích mohlo při UV mapování textur docházet, že na pár místech se textury deformovaly, což bylo v této verzi vyřešeno pomocí rozdělení. GLSL shadery umožňovaly vidět materiály s přiřazenými texturami. Nově se dala v Blender GameEnginu

rozdělovat obrazovku na více pohledů z různých kamer. Tohoto efektu se dalo dosáhnout pomocí propojení Python skriptu druhé či další kamery.(Obr. 5)



Obr. 5: Rozdělení obrazovky a multi-výřezy[6]

1.3.8 Blender 2.44 (Květen 2007)

Verze 2.44 je nyní plně kompatibilní s 64 bitovými procesory a byly přidány nové modifikátory a to „Smooth“ (vyhlazení) a modifikátor „Cast“, který slouží k tvarování objektu.

Změny byly i v GameEngine a Bellet fyzice. Fyzický engine dostal některé změny, které by měly zlepšit přesnost a kvalitu fyzikálních simulací. Vylepšenou přesnost lze aktivovat přes python skripty přes `PhysicConstraints.setNumTimeSubSteps (x)`, kde za hodnotu x se dosazuje přesnost simulace. Čím vyšší hodnota, tím je simulace přesnější. Do herního engine byla přidána nová funkce „Instant Replace Mesh“, která umožňuje nahradit nové Mesh objekty za předešlé, ale se stejným rámem. [6]

1.3.9 Blender 2.46 (Květen 2008)

Tato verze má nová vylepšení systémů díky otevřené projekci filmu „Big Buck Bunny“. Je zde nový částicový systém s nástroji pro rychlou a optimální tvorbu a vykreslování chlupů a kožešiny. Dále je zde vylepšený systém deformací pro pokročilý charakter vybavení a simulace látky a v neposlední řadě se od základu dočkal vylepšení i prohlížeč snímků. Novinkou jsou zde i tzv. měkké stíny, které jsou nyní možné u všech druhů svítidel.

Blender GameEngine dostal spoustu zlepšení s nárůstem play-back rychlosti, množství nových funkcí, včetně filtru 2D kompozic, a samozřejmě s důrazem na kvalitu prostřednictvím oprav chyb.[6]

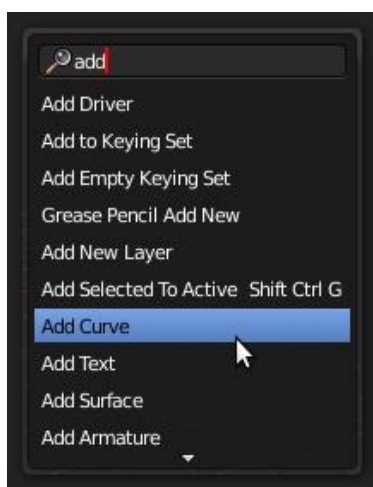
1.3.10 Blender 2.50 (Březen 2010)

Tato verze prošla svou reorganizací a přepsání zdrojového kódu skoro od počátku. Blender dosáhl obrovského zlepšení téměř v každé oblasti, včetně jeho rozhraní, modelování, animačního pracovního postupu, různých nástrojů a Python API. [4] Od této verze se používá Python 3.1.

Uživatelské rozhraní nyní působí přijatelněji, jelikož je možné si nechat zobrazit veškeré funkce a nástroje hned na první pohled, a to bez nutnosti překrývání některého z oken. Nově není také uživatel omezen na práci jen s jednou částí programu, protože v této verzi, již nemusí uživatel doplňovat žádná další data, aby mohl být příkaz dokončen.

V Blenderu 2.5 může být jakákoliv vlastnost zaimprovována, a také si nyní každý uživatel může navolit klávesové zkratky pro každý editor zvlášť. Tento nový systém je nazýván „Animato“.

Další novinkou je integrovaný „Search tool“ (vyhledávač), který se aktivuje v místě, kde chcete hledat stisknutím klávesy „Spacebar“ (mezerník) a zobrazí se menu. Do něj stačí jen zadat název funkce nebo pouze část názvu a automaticky se zobrazují možné výsledky hledání. (Obr. 6) Vylepšení se dočkal také „File Browser“ (prohlížeč souborů), který je nově sjednocen s „Image browserem“ (obrázkovým prohlížečem). Nejen že umožňuje zobrazit seznam souborů, ale nově je také může zobrazovat i jako miniatury. Byl přidán i boční panel, na kterém jsou zobrazeny disky v počítači, naposledy používané soubory, a nově pak každý uživatel si zde může vytvářet i záložky.



Obr. 6: Integrovaný vyhledávač

1.3.11 Blender 2.60 (Říjen 2011)

V této verzi byla do GameEnginu přidána funkce Navigace Mesh objektů, která se používá na zjištění cesty tak, že hráči mohou najít cestu k cíli, nebo mohou utéct pryč. Tato funkce byla implementována pomocí přepracování a obejití knihoven. Nastavení textury obličeje je nově přesunuto do materiálů, kvůli konzistentnosti a snadnější editaci.

Nově byly přidány objekty jako je 3D zvuk a reproduktor, spolu s různými dalšími vylepšeními zvukového systému.[6] Objekt reproduktoru je nyní možno umístit ve scéně, aby bylo možné přivést zvukový soubor do animace.

1.3.12 Blender 2.61 (Prosinec 2011)

V této verzi jsou nejpozoruhodnějšími novinkami například „Render Cycle Engine“, který je k dispozici vedle vnitřního Beder renderu. Je založený na raytracing renderovacím enginu s podporou pro interaktivní rendering. Má v sobě nový systém stínových uzlů, nové textury a akceleraci grafického procesoru (GPU). Tím že se použije GPU rendering namísto klasického, který využívá k výpočtům procesor, se může urychlit vykreslování.

Cycle Tender Engine má dva renderovací módy, přes CUDA, což je preferovaná metoda pro grafické karty Video, tak druhý OpenCL, který je určen pro podporu vykreslování na AMD / ATI grafických kartách.[6] Zatím je to pouze ve fázi experimentu.

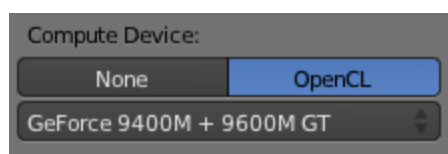
Novinkou je tady i nástroj simulace oceánu, který je formou modifikátoru, který dokáže simulovat a deformovat generovaný povrch oceánu se související texturou, a který nepoužívá pro vykreslení simulačních dat. Další novinkou je fyzikálního systému a modifikátor dynamické barvy, který tímto přináší mnoho možných nových efektů, jež byly dříve skoro nerealizovatelné, například stopy ve sněhu atd.

Jako poslední větší novinkou je to, že v této verzi byla přidána podpora sledování pohybu kamery, s možností rekonstrukce animace z reálných záběrů a kompozitních 3D objektů poskytnutých ve filmových klipech. Byl zde přidán i nový editor pro videosekvenci, sledování bodů a rekonstrukci pohybu. Tato práce je psána pro tuto verzi programu, jelikož to byla aktuální verze ke dni zadání této bakalářské práce.

V průběhu několika měsíců, vyšly i nové verze programu a to konkrétně Blender 2.62 a Blender 2.63

1.3.13 Blender 2.62 (Březen 2012)

V této verzi bylo opraveno mnoho chyb, které byly v předešlé verzi, a došlo tady k několika vylepšením, jako například v Render enginu, kde byly přidány nové funkce, včetně renderovacích vrstev a průchodů, multi GPU renderování a výběru zařízení GPU (Obr. 7), vykreslování hranic objektů a byly zde přidána také nové možnosti pro stínování barvy a uzel pro šachovnicové textury.



Obr. 7: Výběr GPU pro Multi GPU rendering [6]

Nalézá se zde i nový modifikátor, který používá knihovnu pro výřezy, díky které má být možno dosáhnouti mnohem lepších výsledků. Obecné pracovní postupy a možnosti, které jsou k dispozici, jsou v uživatelském rozhraní nezměněny, modifikátor jednoduše pracuje rychleji a vytváří lepší výstupní síť. Jsou zde i změny v chování, jako například „Carve“, což je modifikátor, který provede logickou operaci pouze v té chvíli, je-li spojení dvou mesh objektů uzavřená smyčka hran.[6]

Dalšími novinkami této verze je uživatelské rozhraní integrované části Blender GameEngine, kde panel vlastností byl reorganizován a rozšířen o další relativní možnosti. Je zde přidána možnost, aby Blender převzal rozlišení pracovní plochy hráče a toto rozlišení bylo použito jako „Fullscreen“ rozlišení pro interaktivní aplikaci. Dále se nyní může interaktivní aplikace spouštět v prostředí Blenderu bez nutnosti jejího exportování do spustitelného souboru s koncovkou .EXE, a to vede ke zkrácení času při hledání případných problémů ve vytvořené aplikaci.

V Python API došlo také k několika změnám, a to zejména ve spojení s matematikou. Matice a vektory matematických tříd byly vylepšeny různými způsoby. [6] Byly změněny sloupce a řádky v maticích tak, aby bylo možné k nim rychleji a snadněji přistupovat, a navíc byl dodán i překlad. Nově vektory mohou mít libovolnou velikost a matice jsou lépe vykreslovány na konzole.

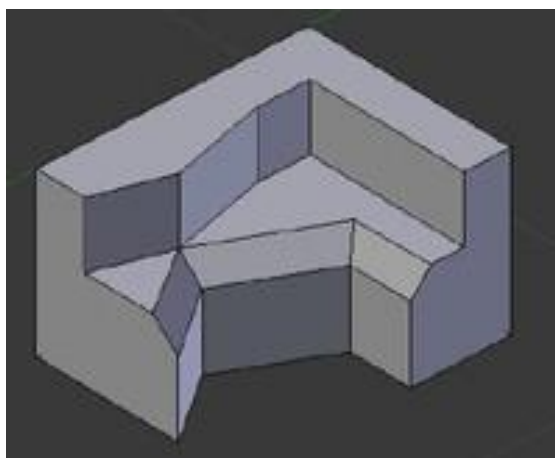
1.3.14 Blender 2.63 (Duben 2012)

Novinkou v této verzi jsou BMesh objekty, které jsou přidány do mesh systému s plnou podporou N-rozměrných polygonů, místo aby byly pouze trojúhelníkové a čtvercové. (Obr. 8) Je tady také nový nástroj, který v editačním režimu, po zvolení ploch, hran nebo bodů je rozpustí.

Další novinkou je podpora pro panoramatické kamery, přesné „float“ (plovoucí) textury a další, jako například nové renderovací vrstvy, vrstvy pro maskování a výřezy z obrázků zobrazených na pozadí a renderovacích vrstvách.

„Motion Tracker“ neboli sledování pohybu dostalo také několik vylepšení, které jsou většinou spojené se stabilizací ve 2D a několik dalších nástrojů. Další zajímavou věcí je „Sculpt Hidding“, díky kterému mohou být části mesh objektů skryty ve Sculpt režimu s cílem zlepšit výkon a vyřezat části mesh objektu, které mohou být obtížně přístupné. [7]

Nově je v této verzi zcela přepracován i nástroj nůž, díky kterému je možné vytvářet libovolné řezy a poté s vybranými plochami i nadále pracovat.



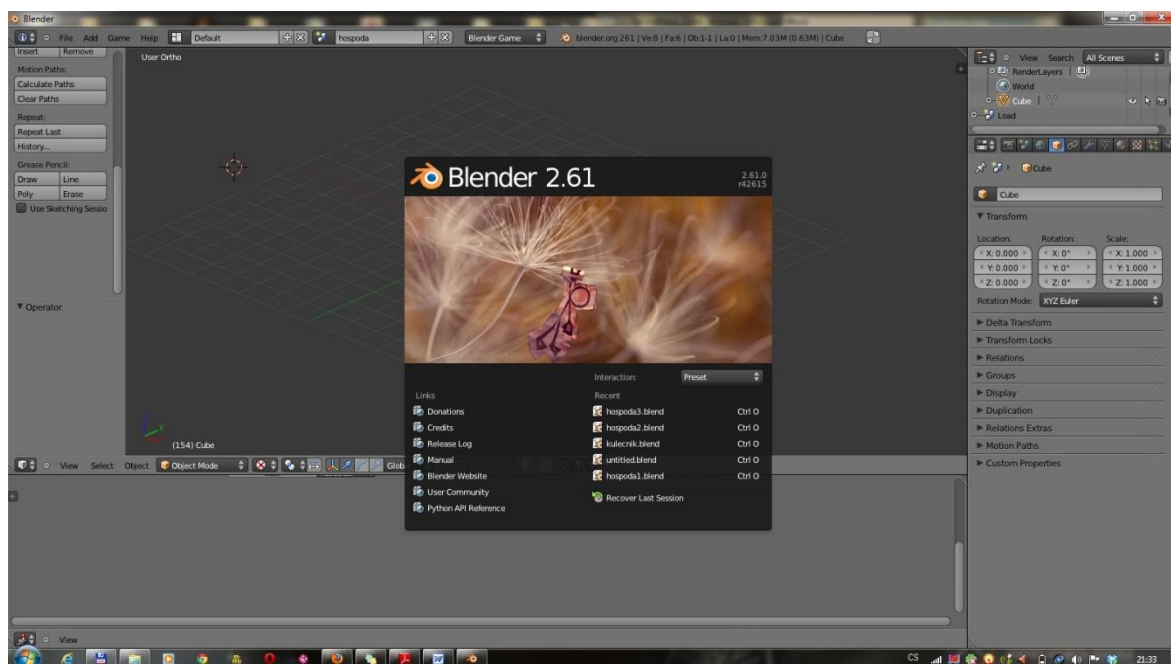
Obr. 8: Příklad BMesh objekt [7]

2 PROSTŘEDÍ PRO TVORBU 3D INTERAKTIVNÍCH APLIKACÍ

Tato kapitola je zaměřena na popis základního prostředí programu Blender pro tvorbu interaktivního obsahu, aplikací a her. Toto prostředí je rozsáhlé, mimo jiné zde mohou být zařazeny například okno pro tvorbu 3D scény (3D view), panel pro tvorbu Herní logiky (Game Logic), okno pro přiladění textur objektům a další okna. Každé z těchto oken bude popsáno níže, kde bude podrobněji popsáno a vysvětleno co se v těchto oknech nachází, k čemu slouží a jak se s nimi pracuje.

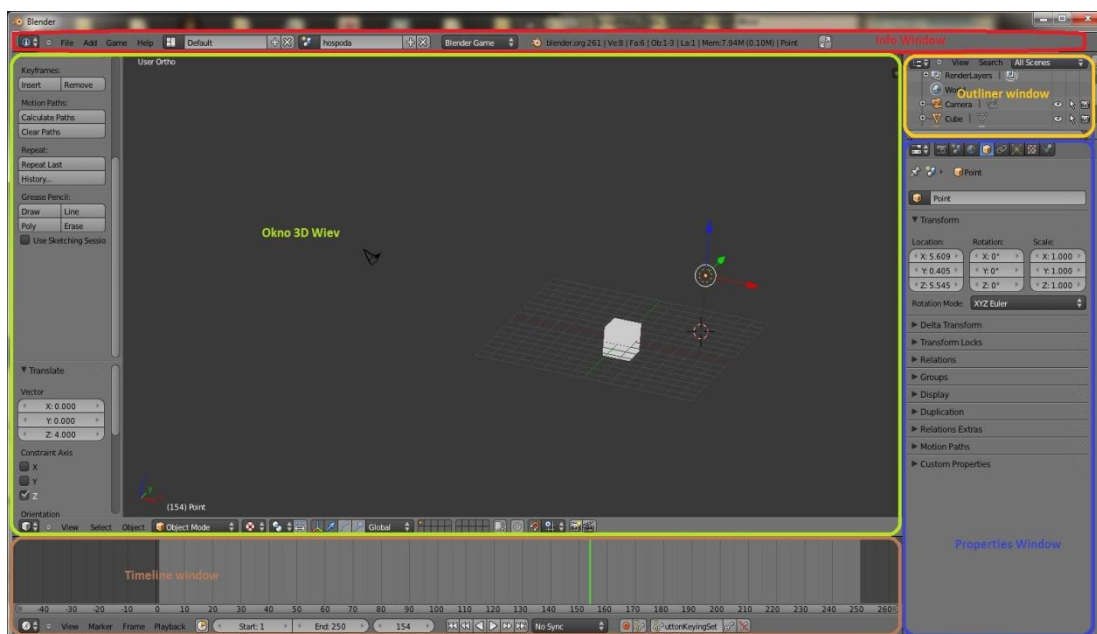
2.1 Základní uživatelské rozhraní programu Blender

Po spuštění programu se vždy zobrazí základní zobrazení s úvodní obrazovkou uprostřed. (Obr. 9) Úvodní obrazovka slouží k tomu, aby bylo možné rychle a snadno přistupovat k naposledy otevřeným souborům. Pokud je potřeba zahájit práci na novém projektu, jsou různé možnosti jak úvodní obrazovku zavřít a to, buď stisknutím klávesy ESC, nebo kliknutím myši mimo úvodní obrazovku.



Obr. 9: Okno programu Blender s úvodní obrazovkou

Po zavření úvodní obrazovky je zobrazena výchozí scéna, ve které se nachází pouze krychle (Cube), světelný zdroj (Lamp) a nakonec kamera. (Obr. 10) Výchozí scéna se načítá při každém spuštění aplikace Blender nebo při vytváření nového souboru.[8]



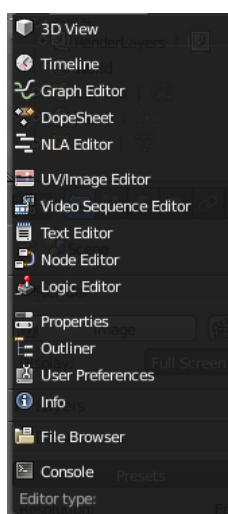
Obr. 10: Výchozí scéna Blenderu

Výchozí obrazovka, jak je možné vidět, je rozdělena do pěti částí, z nichž nejdůležitější je 3D View okno (okno 3D pohledu – žlutě vyznačeno), ve kterém se vytváří a zobrazuje jakýkoliv 3D obsah. V horní části okna programu se nachází Info window (informační okno – červeně vyznačeno), ve kterém jsou rolovací nabídky pro práci s programem. Napravo jsou zobrazována dvě okna, a to Outliner Windows (oranžově vyznačeno), ve kterém jsou zobrazeny všechny objekty použité ve scéně. Je to velmi užitečné, pokud se pracuje s velkým počtem položek. Dalším oknem zobrazovaným vpravo je Properties window (okno vlastností – modře vyznačeno), ve kterém se nachází veškeré možnosti nastavení pro vybraný 3D obsah. Jako poslední je vyobrazeno Timeline window (časové okno – hnědě vyznačeno), které slouží jako časová osa pro animaci.

V každém okně se v pravém horním rohu a levém spodním rohu zobrazuje vyplněný trojúhelník, pomocí něhož je možné rozdělení okna do několika menších oken. To se provádí tak, že se kurzorem myši na něj najede, uchopí se a potáhne se směrem doleva,

pokud je zapotřebí rozdělit okno vertikálně, nebo potažením směrem dolů, pokud je zapotřebí rozdělení okna horizontálně.[8]

Samozřejmě každé okno má své záhlaví, ve kterém se dá zvolit, jaký typ okna bude zobrazeno na místo původního (Obr. 11). Po rozbalení se zobrazí pop-up menu, ve kterém jsou zobrazeny všechny typy oken, podle toho, k čemu slouží. Tady je možno najít i Logic panel (panel Logiky), který slouží k nastavení chování objektů například v interaktivních aplikacích. Tato skutečnost je výhodou pro ty, kterým výchozí rozdělení něčím nevyhovuje, a chtějí si vytvořit své vlastní uspořádání výchozí obrazovky.



Obr. 11: Typy oken

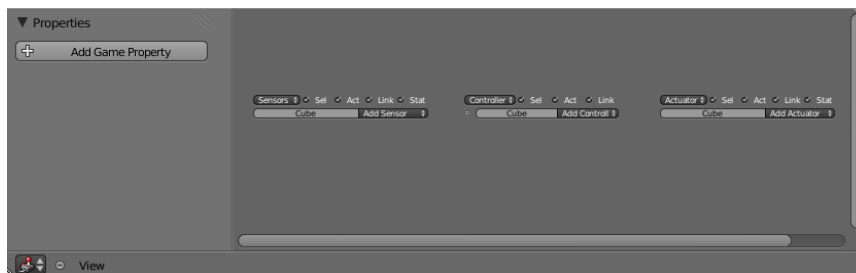
2.2 Game Logic panel

Jak už název panelu napovídá, jedná se o panel pro tvorbu herní logiky, nastavení vlastností chování objektům ve scéně, nastavení čtení vstupů z klávesnice, popřípadě myši či joysticku, propojení s Python API pomocí skriptů, atd.. Tento panel obsahuje tři typy logických bloků. Dalo by se říci, že tyto logické bloky jsou jádrem Blender GameEngine. Cílem logických bloků je nabídnout snadno použitelné rozhraní pro vizuální navrhování interaktivních aplikací bez znalosti programovacího jazyka.[7]

Panel pro tvorbu herní logiky získáme například tak, že na místě Timeline window klikneme na pop-up menu s typy oken a zvolíme položku Game Logic. Zvolené okno se nám zobrazí v dolní části okna programu Blender. Jak již bylo zmíněno výše, jsou tři logické bloky, „Sensors“ (senzory), „Controllers“ (kontroléry) a „Actuators“ (akční členy). (Obr. 12)

2.2.1 Senzory

U senzorů začíná jakákoliv logická akce. Senzory, jakož to smysly objektů reagují na podmínky, v kterém může být stisknutí klávesy, časovaná událost, nebo jiné. Při spuštění senzoru je odeslán impuls do všech připojených kontrolérů, které jsou s aktuálním senzorem spojené. Nový senzor se přidá pomocí tlačítka „Add Sensor“ (přidat senzor), které se nachází vedle názvu objektu, kterému má být daná vlastnost přiřazena.



Obr. 12: Game Logic panel - Panel pro tvorbu herní logiky

2.2.2 Kontroléry

Kontroléry tvoří propojení mezi senzory a akčními členy. Jsou to členy, které shromažďují odeslaná data ze senzorů a na základě Booleovy algebry se rozhodují, jak bude tento vstup zpracován. Existuje osm způsobů, jak zpracovávají vstupní signál.[7] Jsou to kontroléry typu „And“ (A), „OR“ (nebo), „Xor“ (Exkluzivní or), „NAND“ (Negovaný AND), „NOR“ (Negovaný OR), „XNOR“ (Negovaný XOR).

Další dva kontroléry jsou „Expression“ (Výraz), který vyhodnocuje uživatelem napsané vyjádření, což může mít za následek dva typy odpovědi, buď „True“ (pravda, hodnota Log. 1) anebo „False“ (nepravda, hodnota Log. 0). Posledním kontrolérem je „Python Controller“, který kontroluje vstup pomocí uživatelsky naprogramovaných skriptů.

2.2.3 Akční členy

Akční členy provádějí akce, jako jsou například přesun, vytvoření objektu, přehrání zvuku, volba aktivní scény a další. Akční členy zahájí své funkce tehdy, přijde-li na jejich vstup pozitivní impuls od jednoho nebo více kontrolérů, které jsou s daným akčním členem spojeny.[7] Je mnoho typů akčních členů, proto tady budou uvedeny jen některé základní typy, jako jsou například:

Motion nastavuje objektu vlastnosti, jako pohyb a rotace. Existuje mnoho možností přes „teleportaci“, až po fyzické tlačení nebo přesouvání objektu

Sound používá se k přehrání zvuku ve hře

Scene slouží například k nastavení další scény jako aktivní.

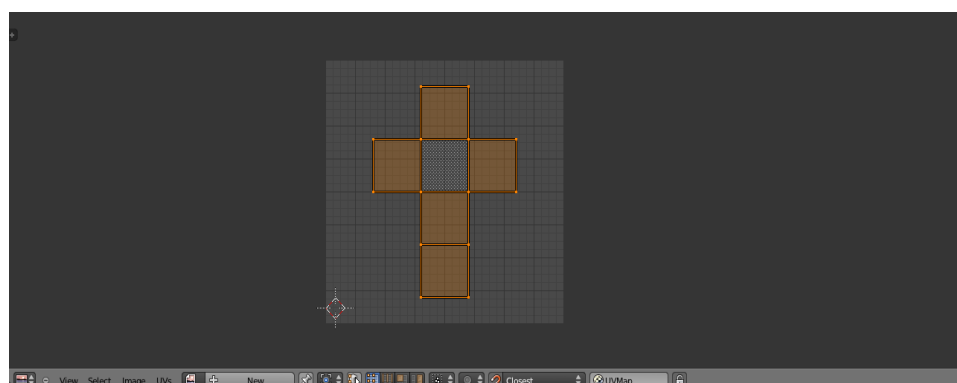
Message odesílá zprávy, které mohou být získané jinými objekty při jejich aktivaci

Podrobnější popis GameLogic panelu bude uveden v Příručce pro tvorbu interaktivních aplikací, kde budou podrobněji popsány nastavení akčních členů, kontrolérů a senzorů, jejich principy a tlačítka se svými funkcemi.

2.3 UV / Image editor

Tento editor je velmi užitečný, pokud se v GameEnginu nezobrazují správně textury. Pomocí UV mapování, se totiž dá dosáhnout výborného výsledku v nanášení požadovaných textur i na složitější materiály. Editor je přístupný jeho vybráním v rozbalovacím menu oken. UV mapování je, dá se říct, standardem pro použití textur na modely, jelikož téměř každý model, který je možno najít ve hře nebo v interaktivní aplikaci je UV mapován.[7] Na objekty v UV/Image editoru je možné používat také transformační příkazy jako v okně 3D View.

V záhlaví okna UV/Image editoru, jsou položky, které slouží k práci s obrázky a jejich nanášení na vybraný model. Na obrázku č. 13 je zobrazena projekce základního Mesh objektu, a to krychle. V záhlaví se nacházejí rozbalovací položky jako je „Select“, které slouží pro volbu výběru buďto bodu, hrany anebo celé plochy. Dále je zde položka Image, která slouží k otevření vybrané textury, a tlačítko Image, které uživateli dovoluje vytvořit si novou texturu.



Obr. 13: UV Projekce základní krychle

II. PRAKTICKÁ ČÁST

3 DOKUMENTACE K VYTVOŘENÉ INTERAKTIVNÍ APLIKACI

Tato kapitola se zabývá tvorbou 3D interaktivních aplikací a her, tudíž jsou zde popsány jednotlivé nástroje pro práci s fyzikálním enginem, tvorbou herní logiky, práci se scénou a vytváření nabídkových menu.

Na přiloženém CD je obsažena názorná ukázka spustitelné interaktivní 3D aplikace s využitím implementované části Blenderu GameEngine (jako soubor s příponou .EXE), kde jsou demonstrovány použité nástroje a funkce, a dále je přiložena elektronická příručka pro tvorbu interaktivních aplikací.

3.1 Vytváření objektů ve scéně

Modely ve scéně tvoří základní kámen a zároveň startovací bod pro tvorbu 3D interaktivních aplikací. Každý objekt, který se nachází ve scéně, byl vymodelován klasickými modelovacími technikami, mezi které patří vytažení (Extrude), rotace, práce s křivkami a textem, rozdělování (Subdivide), a mnoho dalších. Samozřejmě jsou i jiné způsoby jak tvořit objekty, jako například pomocí Python skriptů, nebo generátorů.

Je-li potřeba upravit vložený objekt, je nutno se přepnout pomocí tlačítka TAB na klávesnici do editačního módu objektu, který je vybrán a v tomto okamžiku je možné s ním dále pracovat a uzpůsobovat dle představ každého uživatele dle svých představ.

Při tvorbě interaktivních aplikací je pravidlem, že čím méně vertexů je ve scéně obsažených, tím lépe pro renderování. Jedná se zejména o objekty, které jsou vypočítávány herním enginem, což má za následek, že přebytečné (zbytečné) vertexy zpomalují výpočty grafického a fyzikálního enginu a tím se celá situace stává HW náročnější. Platí zde přímá úměra mezi množstvím ploch a vertexů v objektech, které jsou počítány a rychlostí, jakým probíhají výpočty (například výpočty kolizí, animací, atd.). Aby byla zajištěna co nejkratší doba výpočtů fyzikálního a grafického enginu, dá se řešit problém s přebytečným množstvím vertexů tak, že pomocí tlačítka „Remove Doubles“ v editačním módu objektu, jsou odstraněny nadbytečné (zdvojené) vertexy.

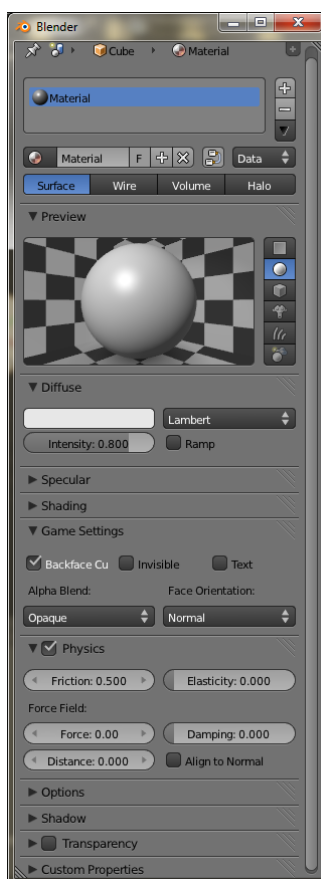
Tato práce je soustředěna na implementovanou část Binderu známou jako GameEngine, proto zde nebudou podrobně popsány techniky tvorby objektů ve scéně, pouze zde budou znázorněny postupy, které jsou potřeba pro správnou funkčnost v aplikaci, a problémy, které mohou nastat při tvorbě interaktivní aplikace jako takové.

3.2 Nastavení materiálů

Pokud je již vytvořený 3D obsah ve scéně, je přiřadit objektům příslušný materiál, který se bude zobrazovat v aplikaci. Je nutné poznamenat, že objekty v aplikaci nebudou nikdy vypadat stejně jako při renderu. Tady může nastat také problém se zobrazováním některých efektů v Game engine, jako jsou například stíny nebo částice. Povrchové vlastnosti a dopadající světlo mohou naznačit, jak moc by se mělo odrazit světlo zpět pod stejným úhlem jako je úhel dopadu (základní princip Render engineu)[9]

3.2.1 Properties

V panelu „Properties“ (vlastnosti) se nacházejí veškeré nástroje potřebné pro přiřazování materiálu, textur a nastavování jejich vlastností. Díky tomu, že veškeré tyto možnosti a volby jsou na jednom místě, dochází ke zpřehlednění a zvyšuje se rychlost práce s 3D obsahem.



Obr. 14: Material Panel

3.2.1.1 Material

Karta „Material“ slouží k vytvoření nového materiálu vybranému objektu, nebo může danému objektu přiřadit již existující materiál. Dále se zde nachází jednotlivá nastavení materiálu, jako je barva, nastavení alfa kanálu, nastavení odrazu světla a další.(Obr. 14)

V horní části tohoto okna se nacházejí tlačítka, které znázorňují možné typy materiálů. Tyto přepínače říkájí Blenderu, kde je vhodný popelíne a jaké aspekty materiálu musí být poskytnuty[9].

Surface vykresluje objekt jako plochy

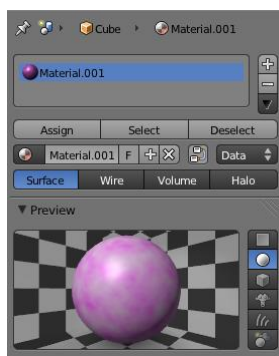
Wire vykresluje hrany a plochy jako dráty (není podporován v Ray tracingu)

Volume vykresluje objekt jako svazek

Halo vykresluje objekt jako Halo částice

V záložce „Deffuse“ se nastavují vlastnosti jako je barva materiálu při dopadu světla pod úhlem a model, který se má použít pro výpočet difúzí.

Další záložkou je záložka „Spectacular“, ve které se dají nastavovat hodnoty stejné hodnoty jako v předešlé záložce, jen s tím rozdílem, že se zde provádějí výpočty pro zrcadlení objektu.



Obr. 15: Tlačítka přiřazení materiálu

Mezi důležité záložky z hlediska pro GameEngine sem patří záložka „Game Setings“, kde se materiálu přiřazují vlastnosti jako neviditelnost, text, orientace ploch a zobrazování zadních ploch v herním enginu., zatímco v záložce „Physics“ se nastavují fyzikální záležitosti, jako jsou například elasticita, odpor, velikost síly, kterou objekt bude působit na ostatní, atd.

Materiály se nejen přiřazují celému objektu, ale je možné přiřadit určitý materiál pouze některým vybraným plochám. Toho lze dosáhnout v Editačním módu objektu, kdy

se vyberou všechny plochy, kterým se má přiřadit nový materiál. Tlačítka, která k tomuto slouží, jsou zobrazena na obrázku 15.

3.2.1.2 Textures

Tato nabídka je určená k přiřazování textur materiálu, které je nanesen na zvolený objekt. V rozbalovacím menu je možné vybírat z mnoha druhů textur, které má v sobě Blender zabudované, od textur Oblaků, až po možnost vložení vlastního obrázku.

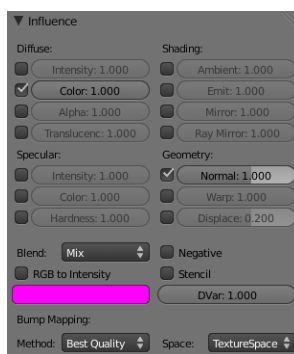
Je ovšem důležité nezapomenout, že textury, se zobrazují rozdílně při renderování scény a v interaktivní aplikaci. Texturu je možné přiřadit jak celému objektu ve scéně, tak i pouze vybraným plochám objektu.

Záložka „Colors“ slouží k nastavení základních vlastností textury, jako jsou například jas, kontrast a sytost textury. Dále se zde dá nastavit barva textury pomocí zadání hodnot pro kód RGB.[8]

V záložce „Mapping“ se nastavují souřadnice, které se mají použít pro mapování textury. Mimo jiné typy souřadnic jsou zde některé, které se používají nejčastěji, jako například:

- Global** používají se globální souřadnice
- Original** používají se původní souřadnice
- Object** souřadnice připojeného objektu
- UV** UV souřadnice pro textury
- Normal** souřadnice normálového vektoru

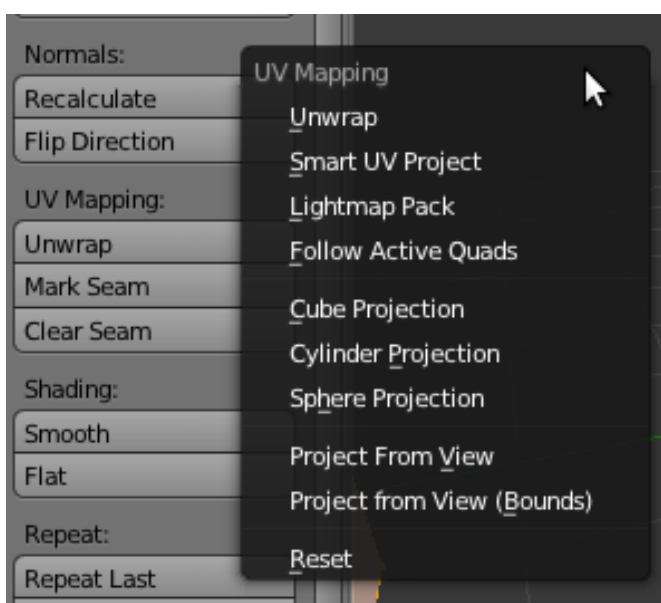
V záložce Influence se nachází rozšířená nastavení pro textury, které určují, jaký vliv bude mít textura na materiál. [7] Záložka je rozdělena na čtyři části a to na „Diffuse“, „Shading“, „Specular“ a „Geometry“. (Obr. 16)



Obr. 16: Záložka Influence

V části „Deffuse“ jsou k dispozici položky jako intenzita, barva, Alfa a průsvitnost. V části „Specular“, která má vliv na zrcadlení textury jsou k výběru tyto možnosti nastavení jako intenzita, barva a tvrdost. Dále je možné zde měnit nastavení textur, které ovlivňují stíny, a to konkrétně v části „Shading“, kde se nastavují hodnoty pro okolí, vyvážení okolního světla, barvu zrcadlení, jako poslední Ray Mirror, které upravuje sílu raytraced zrcadlového odrazu.

Pro správné zobrazování textur v Blender GameEnginu je zapotřebí přemapovat textury pro nanesení na objekt pomocí UV / Image Editoru.



Obr. 17: UV „Mapping Types“

3.2.1.3 UV / Image Editor

Pro mapování textury v UV editoru je nutné vybraný objekt v editačním módu rozbalit pomocí klávesy U, po jejímž stisku se zobrazí pop-up menu s mapovacími typy (Obr. 17). Blender nabízí několik způsobů UV mapování. Jednodušší metody používají vzorce, které mapují 3D prostor do 2D prostoru.[7]

Unwrap vyhlazuje Mech objekty do plochého povrchu pomocí podélných řezů

Cube mapuje síť na plochy krychle, která je poté rozvinuta

Sphere Určeno pro koule a kulovité tvary jako planety, oči atd.

Cylinder vhodné pro projekci válcových ploch

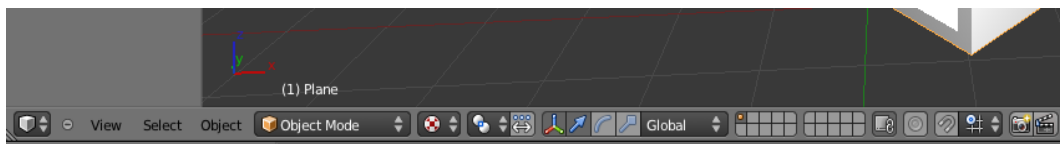
V tomto editoru je nejlepší možností, pokud se volí celé plochy pomocí *Face Select* módu, díky čemuž stačí jen plochy pomocí klasických příkazů transformace přizpůsobovat tak, aby výsledný obraz byl co nejpresnější kopií originálu nebo aby výsledek vypadal reálně.

3.2.2 Postup tvorby 3D obsahu

V této kapitole je popsán postup tvorby 3D, aby vše fungovalo a zobrazovalo se správně v implementované části GameEngine programu Blender, od počátečního přidělování materiálů vybranému objektu až k nanášení textur pomocí UV mapování obsahu tak, aby jej pochopil a dokázal použít i začátečník, který se právě rozhodl s programem Blender začít. Tento konkrétní postup bude demonstrován na objektu grilu, který se nachází ve vytvořené interaktivní aplikaci.

Nejprve se ze základního Mesh objektu pomocí standardních modelovacích technik jako je vytažení, rotace a další vytvořil objekt grilu. Poté byl tomuto objektu přidělen, materiál pomocí karty *Material* v panelu „Properties“. Stisknutím tlačítka *NEW* se vymodelovanému objektu přiřadí základní materiál, který si každý uživatel může pojmenovat dle svého uvážení. Jelikož je objekt složen z jednoho materiálu, stačilo jej přiřadit celému objektu, bez nutnosti vybrání ploch v editačním módu objektu, který se aktivuje stiskem tlačítka *TAB* na klávesnici. Je-li přiřazen základní materiál, nastává otázka správného nastavení.

Nejprve v hlavičce okna 3D View, která se nachází ve spodní části okna, je nutné změnit metodu zobrazování 3D obsahu. Primárně je tato možnost nastavena na hodnotu „Solid“, která zobrazuje obsah bez textur, pouze jako materiál s jemu přiřazenou barvou.(Obr. 18)



Obr. 18: „Header menu“ panelu 3D View

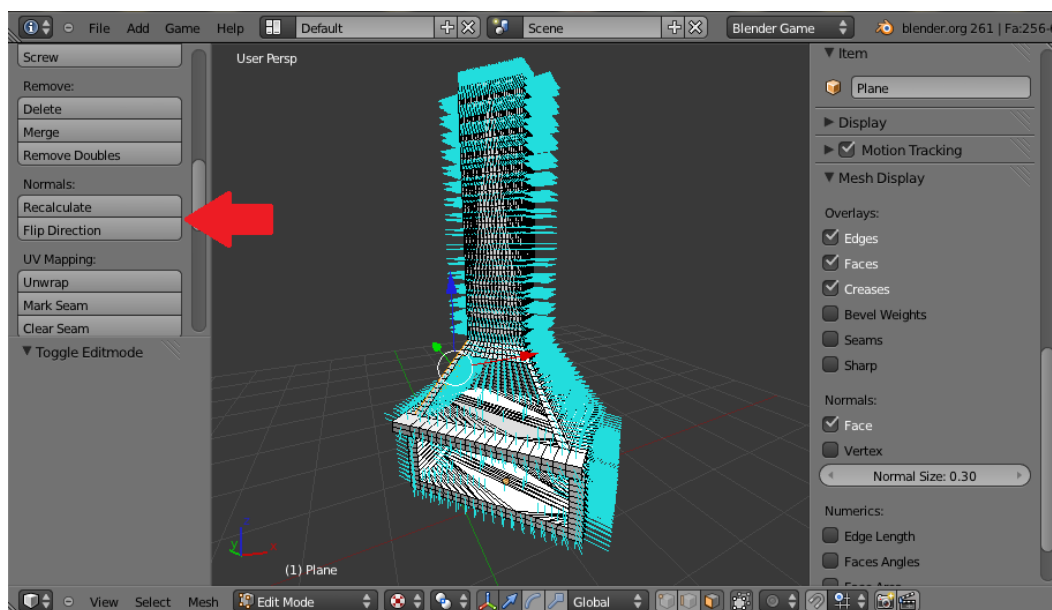
Je-li metoda zobrazování přepnuta správně, je vidět ihned na první pohled, protože je objekt zobrazován s definovanými vlastnostmi.

Dále by bylo dobré nastavit v panelu „Properties“ a záložce „Physics“ ještě fyzikální vlastnosti materiálu, jako jsou například elasticita, velikost síly (Force), kterou bude objekt působit na své okolí, odpor (Distance) a další. Z důvodu toho, že se jedná o objekt, který se ve vytvořené scéně nebude pohybovat, mohou být ponechány veškeré fyzikální hodnoty beze změny. Je ovšem nutno zaškrtnout tuto záložku, jinak nebudou tyto hodnoty brány v úvahu při výpočtech.

Dále je dobré zkontrolovat, aby plochy byly viditelné z venku a ne zevnitř, jak se to může někdy stát. Nejprve je tedy zapotřebí vybrat objekt, u kterého má být kontrola provedena a následně se přepnout v okně 3D View do editačního módu objektu. Stisknutím klávesy N se v pravém okraji zobrazí nové možnosti nastavení objektu.

V tuto chvíli je nejdůležitější záložka „Mesh Display“, ve které jsou nastavení týkající se toho, jak bude vybraný mesh objekt zobrazován v okně 3D View. Po rozevření této nabídky, je nutné v části „Normals“ zaškrtnout volbu *Face*, díky které jsou nyní zobrazeny normály ploch. Pomocí nabídky *Normal Size* si může uživatel zvolit, jak mají být zobrazované normály velké. Kvůli přehlednosti je vhodné volit tuto hodnotu mezi 0,1 až 0,5 u složitějších objektů, jako je například možné vidět na obrázku 19.

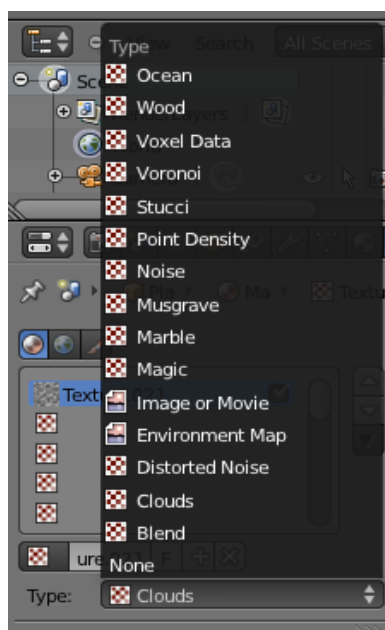
Pokud směřují normály z povrchu objektu směrem ven do prostoru, pak bude textura nanášena na vnější plochy objektu. Nicméně, někdy dochází ke stavu, že normály ploch směřují směrem dovnitř do objektu, což má za následek jedinou věc a to, že přiřazená textura se bude nanášet na vnitřní stranu vymodelovaného objektu. Pokud tento případ nastane, a je nežádoucí, pak jej lze vyřešit pomocí tlačítka *Flip Direction*, které slouží pro práci s normály. Stiskem tohoto tlačítka je vyvolána akce, která přehodí vnitřní normály za vnější a naopak, tudíž se nově textura nanese z vnější strany objektu.



Obr. 19: Kontrola správnosti zobrazování ploch

Pokud již je tedy vše ohledně materiálu nastaveno podle představ uživatele, soustředí se tudíž pozornost na přiřazení textury celému objektu, nebo pouze jeho části.

Nyní je zapotřebí přiřadit novou texturu pro daný objekt. Nová textura se vytváří v panelu „Properties“, kde je nutné přejít na položku „Textures“, která je znázorněná ikonou červeno – bílé šachovnice. Po otevření je zapotřebí vytvořit novou texturu pomocí stlačení tlačítka NEW, podobně jak tomu bylo u tvorby materiálu.

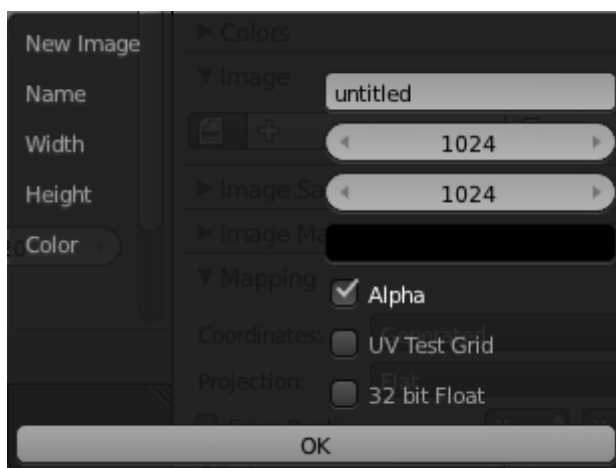


Obr. 20: Volba zobrazované textury

Nyní je vytvořená nová textura, které je nutné ale vybrat její typ, jenž bude zobrazován. Jako typ textury byl zvolen *Image or Movie*, díky kterému je možné přiřadit textuře jakýkoliv obrázek, který uzná uživatel jako žádoucí. Po zvolení této možnosti přibyla níže záložka *Image*, ve které jsou dvě tlačítka. První tlačítko *New* a druhé *Open*. Tlačítko *New* slouží k vytvoření nového obrázku. Po zvolení této možnosti je otevřeno nové okno, ve kterém jsou základní nastavení, které bude uživatelem vytvořený obrázek mít, jako například jméno obrázku, výška, šířka, a další (Obr. 21).

Druhou volbou je možnost kliknutí na tlačítko *Open*, po jehož zvolení je otevřené okno *File browser*, kde si uživatel může najít požadovaný obrázek. Orientace v prohlížeči je velmi snadná, jelikož je uspořádán přehledně. V levém boku je zobrazen panel, na kterém jsou vyobrazeny disky počítače, záložky a v neposlední řadě také často používané složky. Ve vrchní části si uživatel pomocí volby ikon může sám rozhodnout, jaký typ souborů se mu bude zobrazovat, a které mu zůstane „skrytý“.

Jsou zde vidět ještě dvě textová pole, jež vrchní obsahuje cestu k souboru a říká uživateli, kde se právě nachází, a druhé pole slouží k zadání názvu obrázku. Toto pole se automaticky vyplní, pokud uživatel najede kurzorem myši na obrázek a zvolí jej stiskem LMB. Pro celkové zvolení obrázku je nutné kliknout na tlačítko *Open Image*, které se nachází napravo ve vrchní části okna.



Obr. 21: Nabídka New Image

Nyní je čas přesunout pozornost na záložku *Mapping*, kde se nastavují vlastnosti, jako například projekce, která nabízí čtyři druhy, „Flat“ (pro projekci ploch), „Cube“ (krychle), „Sphere“ (vhodná pro projekci na objekty kulovitého tvaru) a nakonec „Tube“, která je vhodná pro projekce na objekt válcovitého tvaru. Dále je zde možnost „Coordinate“, která

nastavuje práci se souřadnicemi. Tento postup je vhodný pouze pro správné zobrazování v renderovaných obrázcích a animacích. Pro správné zobrazování textury v herním enginu, je zapotřebí zvolit možnost UV, což zároveň znamená, že je nutno vybráný objekt, na který má být textura nanесena rozbalit pomocí metody *Unwrap*, v UV / Image editoru.

V okně 3D View je zapotřebí vybrat daný objekt, na jehož UV mapu má být nanесena textura. Je-li objekt vybrán, pak po přepnutí do editačního módu, je nutné vybrat všechny plochy, které mají být rozbaleny a namapovány pomocí UV souřadnic. Plochy mohou být vybrány pomocí klávesy A (všechny plochy objektu), klávesou B (s následným držením LMB a tažením je vybráno více ploch) nebo postupným výběrem ploch (stisknutá klávesa SHIFT + LMB pro výběr plochy). Po vybrání všech ploch byla vyvolána nabídka *UV Mapping* pomocí stisknutí klávesy U, ve které se nachází mnoho možností rozbalení objektu. Protože gril je v základu objekt krychlového tvaru, byla vybrána pro výpočet souřadnic možnost *Cube projection*.

V této chvíli je na řadě vybrání textury, která bude přiřazena vypočítaným UV souřadnicím. Toho lze dosáhnout tím, že v záhlaví okna UV / Image editoru vybereme možnost Image (Obr. 22).



Obr. 22: Nabídka Image „UV / Image“ Editoru

Pomocí volby Open Image uživatel vybere v prohlížeči souborů obrázků, který chce, aby byl použit jako textura na vybrané plochy a jak je zřejmé z obrázku 22, byla zvolena

textura bílých cihel[11]. S vybranými plochami v Uv / Image editoru je možné pracovat standardními příkazy pro transformaci s objektem jako je například rotace, posun a změna velikosti. Výsledek, jakého je dosaženo pomocí UV souřadnic, je viditelný již při transformaci ploch.

3.2.3 Vytváření herní logiky a kolizí

Tato podkapitola naznačuje postup při tvorbě herní logiky pro interaktivní aplikace, která určuje chování objektů, aby celá scéna vypadala co možná nejrealističtěji. Všechna nastavení, která jsou potřebná pro tvorbu herní logiky, se nachází v okně „Game Logic“ (Obr. 12). Tady jsou také prováděna všechna nastavení všech objektů, které mají mít určeny nějaký typ speciálního chování, ať už jde o pouhý pohyb, rotaci neb i volbu aktivní scény. Herní logika je základním nástrojem, na kterém stojí celý účel interaktivních aplikací. Tento postup je znázorněn na dvou objektech.

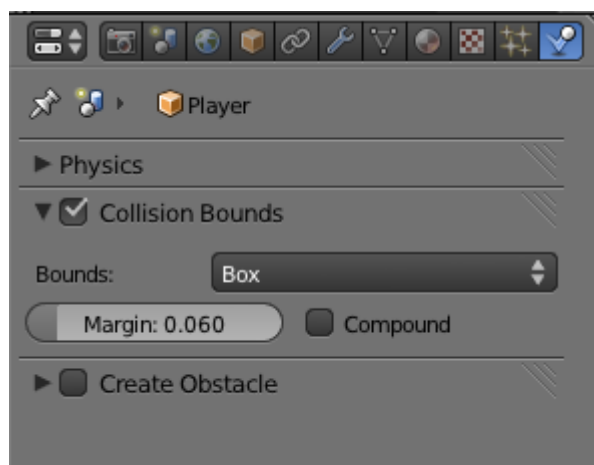
3.2.3.1 Objekt Player

Na objektu Player, která představuje hráče a bude na něm popsána jeho herní logika, přiřazené senzory, kontroléry a akční členy, vysvětlena jejich funkce, a také bude popsáno jeho nastavení Fyziky. Na druhém objektu (Menu) bude popsána ve stručnosti jeho tvorba a herní logika.

Objekt Player (hráč) je základní mesh objekt tvaru krychle, který byl pomocí základních příkazů transformace uzpůsoben do požadovaného tvaru a velikosti. V panelu „Properties“ se nachází záložka *Physics*, ve které je nutno nastavit fyzikální vlastnosti, kterými se bude objekt v interaktivní aplikaci řídit.

Nejprve je nutné nastavit fyzický typ jako *Dynamic*, který zajišťuje, že se objekt může ve scéně pohybovat. Dále zaškrtnutím položky *Actor* je zajištěno, že bude objekt počítán herním enginem. Objekt má fyzický typ *Dynamic*, což zajišťuje, že se objekt řídí fyzikálními zákony a vrací kolize, které je ale nutné nejdříve nastavit v záložce *Collision Bounds* (Obr. 23), kde je více možností, podle kterých se bude objekt řídit kolizemi, například nastavení ohraničení kolizního objektu. Pro objekt hráče, jelikož je kvádrovitého tvaru, bylo ohraničení pro detekci kolizí nastaveno jako *Box*. Dále bylo nutné nastavit velikost hraničního objektu, což se provádí pomocí volby hodnoty *Margin*, její rozmezí se pohybuje od hodnoty 0 do hodnoty 1. Další věcí je možnost, aby byl objekt hráče

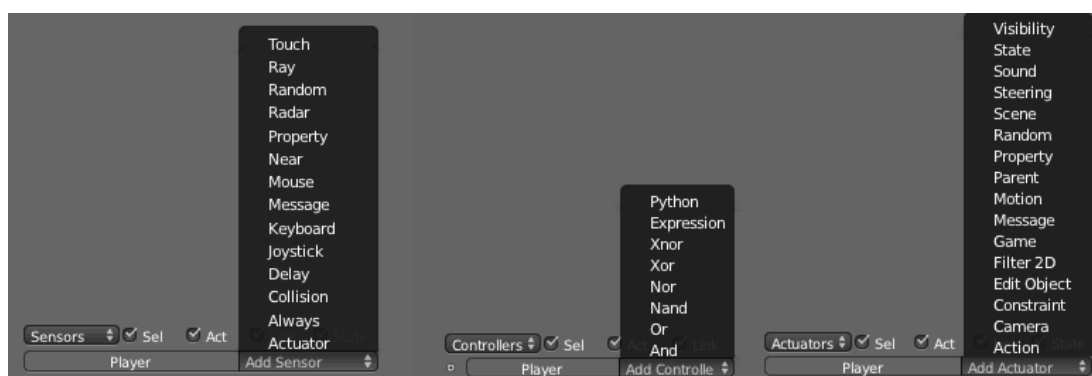
neviditelný (položka *Invisible*). Pokud je položka zaškrtnuta, objekt se nezobrazuje v interaktivní aplikaci.



Obr. 23: Nastavení kolizních objektů

Nyní je na řadě udělat rozhodnutí, co všechno má objekt hráče v aplikaci umět a jaké budou mít tyto vlastnosti následky, jakožto kompletní herní logika (Obr. 25). K tomu jsou v Game Engineu tři logické bloky: Senzory, Kontroléry a Akční členy.

Senzory slouží k tomu, aby zaznamenávaly vstupní signál, který má každá sensor nastavený, například z klávesnice nebo časovačů. Senzory, kontroléry a akční členy jsou různých typů podle toho, z jakého zdroje jim má přicházet vstupní signál, jak mají rozhodnout o zpracování signálu a jakou akci mají provést (Obr. 24).



Obr. 24: Typy senzorů, kontrolérů a akčních členů

Hlavní vlastností, která je u objektu hráče požadována, je pohyblivost. Díky tomu, že je objekt nastaven jako dynamický, stačí tudíž vytvořit senzory, které budou “sbírat“

definovaný vstupní signál, na jehož základě pomocí kontrolérů, které se rozhodují podle výroků, jež mají nastaveny, se signál zpracovává a posílá do akčního členu, jež určuje prováděnou akci.

Je požadováno, aby se objekt pohyboval na základě stisknutých kláves na klávesnici, takže zde bude napsán postup pro jeden senzor z klávesnice. Například u objektu hráče, nacházejícího se v interaktivní aplikaci je pohyb zajištěn pomocí kláves na numerické klávesnici.

Například pohyb vpřed je požadovaný po stisku klávesy 8 na numerické klávesnici. Zvolený typ senzoru, který bude zpracovávat vstupní signál z této klávesy je tudíž typ *Keyboard*, který si uživatel může pojmenovat vložením textu do pole *Sensor Name*, například “vpřed“. Je nutné dále definovat akční klávesu kliknutím myši na prázdné pole vedle textu *Key* a následným zmáčknutím klávesy 8 na numerické klávesnici, a tím je zajištěno, že nyní tento senzor bude zpracovávat pouze signál z klávesy NUM 8.

Následně se přiřadí senzoru kontrolér *And*, který předá vstupní signál akčnímu členu, pokud je vyhodnocovaná podmínka pravdivá. Jako poslední věc, která chybí k tomu, aby už objekt hráče se dokázal pohybovat dopředu, je přidání “koncového“ akčního členu. Tento akční člen je zvolený typu *Motion*, který slouží k nastavení lokace a rotace objektu. Je nutné nastavit příslušnou hodnotu do jednoho ze tří polí (podle souřadnicových os), která jsou učená pro určení lokace objektu. Pohyb vpřed v této aplikaci znamená pohyb po ose Y v kladném směru, tudíž je nutné zadat do prostředního pole pro lokaci hodnotu kroku. Byla zadána hodnota 0,05 z důvodu, aby pohyb nebyl příliš rychlý a těžce ovladatelný. Tento postup je nutné opakovat pro každý pohyb. Rotace objektu se provádí totožným způsobem, pouze s tím rozdílem, že se zadávají hodnoty do jednoho ze tří polí určených pro rotaci objektu (většinou osa Z).

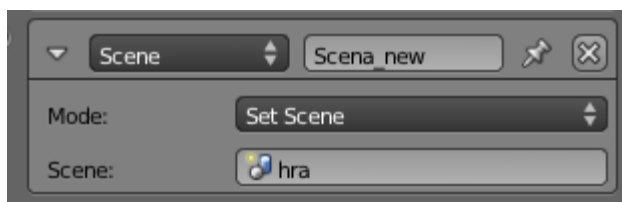


Obr. 25: Herní logika pro objekt hráče

Protože tvorba herní logiky je v programu Blender bez nutnosti znalosti programovacího jazyku, je nutné již pouze tyto tři logické bloky spojit. K tomuto účelu jsou u každého logického bloku příslušná “kolečka“, pokud na ně uživatel klikne a drží LMB, pak přesunutím kurzoru myši na další kolečko dojde ke spojení těchto logických bloků, což je zároveň podpořeno vizualizací spojnice.

Nyní by bylo dobré vysvětlit i možnost, kdy je požadováno například zapnutí hry v nové scéně, při vzniku situace, kdy dojde ke vzájemné kolizi mezi dvěma objekty. K tomuto účelu slouží senzor *Touch*, kterému je vyslán pozitivní impuls jen tehdy, dojde-li ke kolizi daného objektu, jenž má senzor přiřazen s materiálem, který je nastaven jako kolizní v nastavení tohoto senzoru.

V případě této vytvořené aplikace se jedná o situaci, kdy dojde ke kolizi objektu hráče a objektu bludiště (Material.008) ve scéně *hospoda*. K tomuto senzoru je přes připojený kontrolér *And* napojen i akční člen, tentokrát typu *Scene* (Obr. 26). V tomto akčním členu je na výběr mnoho možností nastavení módu, kde nyní je zapotřebí v tomto okamžiku zvolit mód *Set Scene*, který zajistí přepnutí aktuální scény, ve které se hráč nachází, za novou scénu, jenž je v tomto případě scéna *hra*, jejíž název se musí zadat do pole *Scene*.

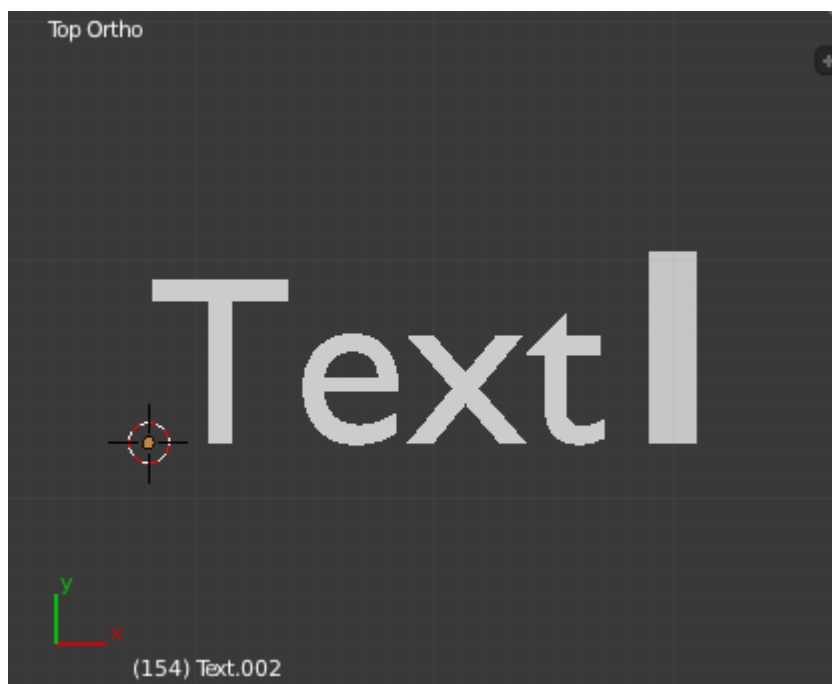


Obr. 26: Akční člen typu „Scene“

3.2.3.2 Menu

Scéna Menu je vytvořena z důvodu toho, aby byla v aplikaci vložena “křížovátka“, pomocí které si uživatel může vybírat jednu z možností, podle toho, co chce udělat. V celé scéně se nacházejí pouze čtyři objekty, přičemž hlavním objektem, na který je soustředěna pozornost, je *Text*.

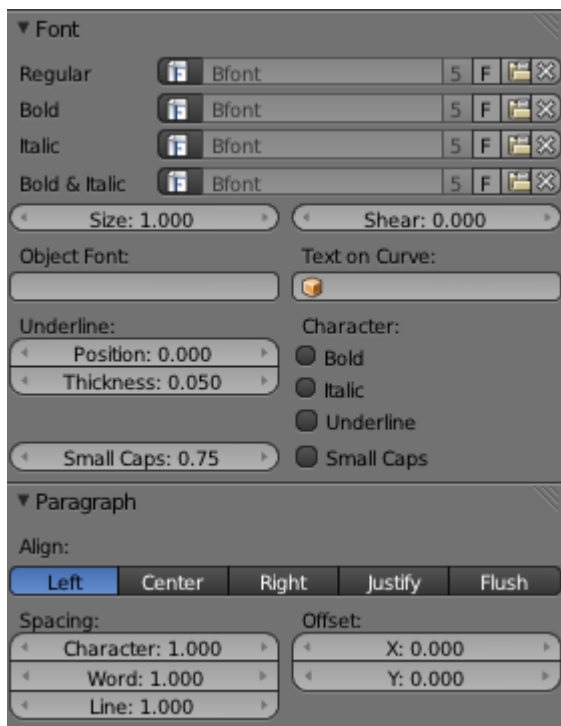
Text, který tvoří tento objekt, je možné změnit v editačním módu (Obr. 27) tohoto objektu, kde je prováděna změna textu jako při psaní v textovém editoru. To znamená, že se zobrazují písmena podle toho, jaké klávesy jsou stisknuty na klávesnici a také zde fungují klávesy Enter a Backspace.



Obr. 27: Editační mód objektu Text

Je-li napsán veškerý text, který má být zobrazován, je nutné nastavit jeho vlastnosti v panelu *Properties*. Pokud je zapnut editační mód objektu v okně 3D View, potom v panelu vlastností je nově zobrazena další karta „Object Data“, ve které se dají nastavit možnosti zobrazovaného textu.

V záložce *Font* jsou k dispozici nastavení vzhledu textu a jeho velikost. Dále je možné nastavovat různé fonty pro různé typy písma, jako je „Bold“ (tučné), „Italic“ (kurzíva) nebo „Underline“ (podtržené). Dále je možné nastavovat vlastnosti jako zarovnání textu, k čemuž složí v záložce *Paragraph* pět tlačítek „Left“ (zarovnat vlevo), „Center“ (zarovnat na střed), „Right“ (zarovnat vpravo) a další (Obr. 28).



Obr. 28: Panel Properties, záložky Font a Paragraph pro nastavení textu

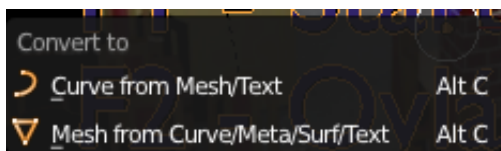
Objektu *Text* bylo dáno zarovnání na střed. Poté bylo již možné opustit editační mód a pomocí stisknutí kláves ALT + C je zapotřebí konvertovat objekt Text na Mesh objekt („Convert Mesh from Curve / Meta / Surf / Text“). (Obr. 29) Teď je nutné pomocí vytažení dát text do 3D, čehož lze dosáhnout v editačním módu Mesh objektu, a následně byla tomuto objektu přiřazena barva materiálu.

Jako poslední přibyla objektu herní logika, ve které je pomocí logických bloků zajištěno výsledné chování objektu, jak je vidět ve vytvořené interaktivní aplikaci. Například rotace je způsobena akčním členem *Motion*, kterému byla nastavena hodnota 1 pro rotaci v ose Y. Proto, aby tento akční člen mohl fungovat stále, musí být zvolen senzor *Always*, který je nastaven na „True Level Triggering“, jenž vysílá pulzy s hodnotou Log. 1 (Hodnota TRUE). (Obr. 30)



Obr. 29: Senzor typu „Always“

Další funkci, která má být zajištěna je ta, že je nutné po stisku tlačítka F1 zapnout hry, což je zajištěno senzorem Keyboard, kterému je přiřazeno příslušné nastavení (bylo zmíněno v podkapitole 3.2.3.1) a který je přes kontrolér *And* spojen s akčním členem typu *Scene*, kde je jako mód vybrána možnost *Set Scene*, a v poli scény je napsáno jméno scény, která má být nově nastavena jako výchozí.



Obr. 30: Metody konvertování

Jako předposlední logický blok, na který by bylo dobré upozornit, je akční člen typu *Game*. Tento akční člen se stará o interaktivní aplikaci jako celek. Na výběr jsou tady možnosti akcí, jako jsou například „Quit Game“ (Opuštění hry), „Restart Game“ (znovu spuštění hry) nebo „Start Game from File“, což umožňuje spustit hru z externího souboru.

Posledním akčním členem je typ *Sound* (Obr. 31), který umožňuje přehrávat jakýkoliv zvukový soubor, který si uživatel vybere, například .mp3 soubor. Dále je zde možnost přehrávání zvuku ve 3D prostoru, jenž zajistíme zaškrtnutím políčka 3D Sound. Dále je zde možnost nastavení hlasitosti („Volume“), jež nabývá hodnot 0 – 2, nebo možnost „Pitch“, která slouží k určení rychlosti přehrávání.



Obr. 31: Akční člen typu „Sound“

Na internetu je k maní mnoho užitečných textur ([11], [12], [13]), které byly použity ve vytvořené interaktivní aplikaci.

ZÁVĚR

Teoretická část této bakalářské práce je rozdělena na dvě části, kde první je soustředěna hlavně na historický vývoj programu, kde jsou uvedeny historické verze spolu se svými novinkami a vylepšeními, jež přinesly oproti předešlým verzím. Dále jsou zde uvedeny přednosti programu Blender, jeho hlavní rysy a charakteristika.

V druhé části je popsáno základní prostředí programu Blender, kde jde zejména o okno „3D View“ pro zobrazení 3D obsahu a modelování scén, panel „Properties“, ve kterém se nacházejí všechny možnosti pro přiřazování materiálu, textura a nastavení fyziky, dále například okno „UV / Image Editor“ pro nanášení textur na objekt, který byl pomocí tohoto okna mapován pomocí UV souřadnic. Za nejdůležitější panel pro tvorbu interaktivních aplikací je považován panel „Game Logic“, díky kterému je možné tvořit herní logiku, jež zajišťuje požadované chování objektů.

Praktická část je zaměřena na popis tvorby vytvořené názorné 3D interaktivní aplikace, kde je pozornost zaměřena především na správné zobrazování textur v GameEnginu, tvorbu herní logiky a práci s fyzikálním enginem. Tyto funkce a nástroje jsou demonstrovány také na vytvořené interaktivní aplikaci.

Jako první výstup této bakalářské práce je vytvořená Příručka pro tvorbu interaktivních aplikací v prostředí programu Blender. Tato příručka je soustředěna především na oblast tvorby herní logiky, jakožto nejdůležitější nástroj pro interaktivní aplikace. Jsou zde podrobně popsány nástroje a funkce logických bloků a popis jejich vlastností. Tato příručka byla tvořena tak, aby jí bylo možné použít jako pomůcku pro výuku programu Blender, především části GameBlender, která se zabývá tvorbou interaktivního 3D obsahu.

Druhým výstupem je vytvořená interaktivní 3D aplikace, ve které jsou demonstrovány schopnosti programu Blender. Použité funkce, nástroje a postup její tvorby je popsán v praktické části této bakalářské práce, kde jsou zmíněny i nastavení fyzikálního enginu.

Všechny výstupy jsou shromážděny na CD přiloženém k této bakalářské práci. Bakalářská práce a příručka pro tvorbu interaktivních aplikací jsou zde uloženy jako soubory ve formátu .PDF a dále je na přiloženém CD vložena i vytvořená interaktivní aplikace.

CONCLUSION

The theoretical part of the Bachelor thesis is divided into two parts. The first part concentrates mainly on the historical development of the program. It includes all historical version of Blender program mentioning also the innovations and improvements which occurred during the time. Moreover, this part discusses main characteristics of Blender program.

The second part describes the basic environment of Blender. This includes mainly the “3D View” window, which enables modeling of scenes, and toolbar “Properties” which offers possibilities for assigning materials, textures and physics setting. Another property discussed in this part is for example “UV / Image Editor” window, which enables applying of textures to an object that has been mapped. The most important toolbar for creating interactive applications is the “Game Logic”. This enables creation of game logic, which ensures the desired conduct of objects.

The practical part focuses on the description of illustrative creation of interactive 3D applications. The main focus is being put on the correct display of textures in GameEngine, on creating of game logic and on work with the physics engine. These features and tools are demonstrated also on created interactive application.

The first output of this work is a guide for building interactive applications in Blender. This brochure mainly concentrates on the area of game logic since this is the most important tool for interactive applications. It includes a detailed description of tools and functions of logic blocks. This guide was created for pedagogy purposes. It should be a useful help when teaching the Bender program (especially GameBlender which deals with the creation of interactive 3D content).

The second output is an interactive 3D application itself, which is demonstrating abilities of Blender software. All functions, tools and processes which were used for creating are described in the practical part of the Bachelor thesis. The practical part also includes the setting of the physical engine.

All outputs are collected on the CD attached to this thesis. Bachelor thesis and guide for the creation of interactive 3D applications are saved in the PDF format. The CD also contains the created interactive 3D application.

SEZNAM POUŽITÉ LITERATURY

- [1] *Charakteristika programu Blender* [online]. [cit. 2012-04-15]. Dostupné z:
< <http://www.blender3d.cz/drupal/?q=charakteristika> >
- [2] POKORNÝ PAVEL. *Blender - naučte se 3D grafiku*. 2.vydání, Praha BEN
- technická literatura, 2009. ISBN 80-7300-244-2.
- [3] *Doc:CZ/2.6/Manual/Introduction/History* [online]. [cit. 2012-04-24]. Dostupné z:
<<http://wiki.blender.org/index.php/Doc:CZ/2.6/Manual/Introduction/History>>
- [4] *Doc:CZ/2.6/Manual/IBlender 2.5 Changes* [online]. [cit. 2012-05-08]. Dostupné
z: http://wiki.blender.org/index.php/Doc:CZ/2.6/Manual/Blender_2.5_Changes
- [5] KLÁSEK, Ivo. *GAMEBLENDER - Tvorba interaktivních aplikací*. Zlín, 2006.
Bakalářská práce. FAI, Univerzita Tomáše Bati.
- [6] *blender.org – Release Logs* [online]. [cit. 2012-05-13]. Dostupné z:
< <http://www.blender.org/development/release-logs/> >
- [7] *Doc:CZ/2.6/Manual - BlenderWiki* [online]. [cit. 2012-05-21]. Dostupné z:
< <http://wiki.blender.org/index.php/Doc:2.6/Manual> >
- [8] *Doc:CZ/2.6/Manual/Interface/Window system* [online]. [cit. 2012-05-28].
Dostupné z:
http://wiki.blender.org/index.php/Doc:2.6/Manual/Interface/Window_system
- [9] *Doc:CZ/2.6/Manual/Materials - BlenderWiki* [online]. [cit. 2012-05-31].
Dostupné z: <<http://wiki.blender.org/index.php/Doc:2.6/Manual/Materials>>
- [10] *Game Engine* [online]. Dostupné z:
<<http://blenderartists.org/forum/forumdisplay.php?34-Game-Engine>>
- [11] *Less about me, and more about school* [online]. Dostupné z:
<http://archinect.com/blog/article/21450955/less-about-me-and-more-about-school>
- [12] *Texture: beton* [online]. Dostupné z:
<<http://letrefle.deviantart.com/art/Texture-beton-161405418>>
- [13] *Textury: Vzorek Střešní Tašky* [online]. Dostupné z:
<http://cz.123rf.com/photo_7517018_textury-vzorek-sta-ea-na-taa-ky.html>
- [14] *Grafika online. Animace v Blenderu. Blender [Online]. 2011.* Dostupné z:
[<http://www.grafika.cz/art/3d/clanek2047803217.html>]

SEZNAM POUŽITÝCH SYMBOLŮ A ZKRATEK

3D	Neboli trojrozměrný (například prostor, zvuk).
Action	Neboli akce, jeden ze akčních členů pro tvorbu herní logiky.
Actuator	Akční člen – vykonává akci (například svaly).
Bullet	Jeden z typů fyziky pro výpočet kolizí a chování objektů ve scéně.
Controller	Neboli kontrolér – zajišťuje funkčnost herní logiky
Driver	Ovladač - z anglického slova „Drive“ neboli ovládat, řídit
FALSE	Odpověď typu nepravda – podmínka akce není splněna
Frame Rate	Obnovovací frekvence (vnitřní čas systému)
Game Engine	Neboli herní engine – určen pro tvorbu a přehrání interaktivních aplikací
GPU	Grafický procesor – pomáhá centrálnímu procesoru při graf. výpočtech
LMB	Neboli levé tlačítko myši – z anglického „Left Mouse Button“
Plug-in	Nebolí zásuvný doplněk – nejčastěji pro internetové prohlížeče
Pop-up	Rozbalovací menu
Real-time	Neboli reálný čas (např. interaktivní aplikace fungující v reálném čase)
Render	Výpočty potřebné k zobrazení obrázku
RMB	Pravé tlačítko myši – z anglického „Right Mouse Buton“
Sensor	Neboli senzor – přijímá signál ze vstupních zařízení
Sound	Neboli zvuk – akční člen slouží k přehrání audio souboru v aplikaci
TRUE	Odpověď typu Pravda – podmínka akce splněna, vyslán pozitivní impulz
UV Calculation	Metody výpočtu UV souřadnic pro mapování textur
Unwrap	Rozvinutí 3D sítě Mesh objektu do 2D roviny
Vertex	Geometrický vrchol objektu

SEZNAM OBRÁZKŮ

Obr. 1: Karta UV kalkulace	15
Obr. 2: Průhledné menu[6]	16
Obr. 3: Hodnoty pro Joystick klobouky [6]	17
Obr. 4: Nastavení objektu jako Herec[5]	18
Obr. 5: Rozdělení obrazovky a multi-výřezy[6]	19
Obr. 6: Integrovaný vyhledávač.....	20
Obr. 7: Výběr GPU pro Multi GPU rendering [6]’	22
Obr. 8: Příklad BMesh objekt [7]	23
Obr. 9: Okno programu Blender s úvodní obrazovkou	24
Obr. 10: Výchozí scéna Blenderu	25
Obr. 11: Typy oken.....	26
Obr. 12: Game Logic panel - Panel pro tvorbu herní logiky	27
Obr. 13: UV Projekce základní krychle.....	28
Obr. 14: Material Panel.....	31
Obr. 15: Tlačítka přiřazení materiálu.....	32
Obr. 16: Záložka Influence	33
Obr. 17: UV „Mapping Types“	34
Obr. 18: „Header menu“ panelu 3D View	35
Obr. 19: Kontrola správnosti zobrazování ploch	37
Obr. 20: Volba zobrazované textury	37
Obr. 21: Nabídka New Image	38
Obr. 22: Nabídka Image „UV / Image“ Editoru	39
Obr. 23: Nastavení kolizních objektů	41
Obr. 24: Typy senzorů, kontrolérů a akčních členů.....	41
Obr. 25: Herní logika pro objekt hráče	42
Obr. 26: Akční člen typu „Scene“	43
Obr. 27: Editační mód objektu Text	44
Obr. 28: Panel Properties, záložky Font a Paragraph pro nastavení textu.....	45
Obr. 29: Senzor typu „Always“	45
Obr. 30: Metody konvertování.....	46
Obr. 31: Akční člen typu „Sound“	46
Obr. 32: Menu z aplikace.....	55

Obr. 33: Hra v aplikaci	56
Obr. 34: Scéna hospoda	57
Obr. 35: Scéna Menu	57

SEZNAM PŘÍLOH

Příloha PI – obsah přiloženého CD

Příloha PII – Obrázky z přiložené interaktivní aplikace

Příloha PIII – Obrázky scén z programu Blender

PŘÍLOHA P I: NÁZEV PŘÍLOHY

Složka s vyrenderovanými obrázky

Složka s texturami

Složka s animacemi z projektu

Soubor projektu hospoda v programu Blender

Složka s vytvořenou aplikací hospoda.exe spolu s potřebnými knihovnami, nutnými k jejímu spuštění

Soubor .zip s verzí programu Blender 2.61, ve které je aplikace vytvořena, a pro niž je tato bakalářská práce a příručka pro tvorbu interaktivních aplikací napsána

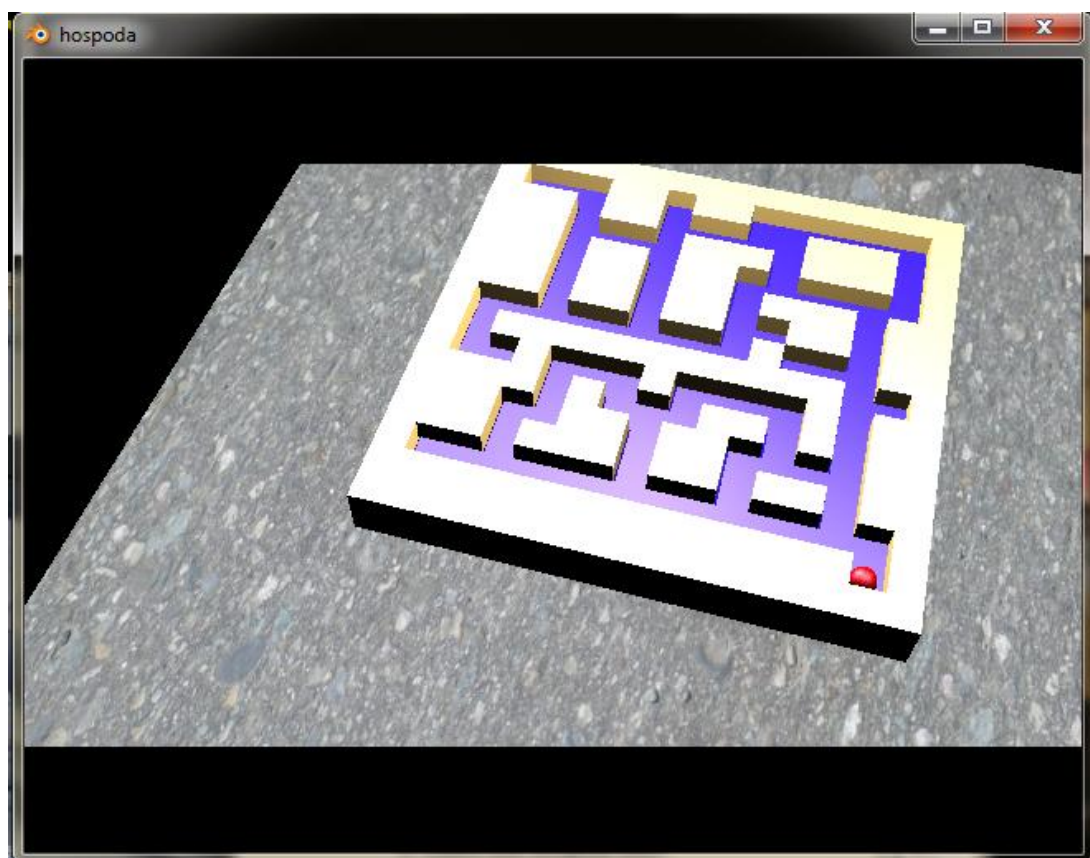
Elektronická příručka pro tvorbu 3D interaktivních aplikací

Elektronická forma bakalářské práce

PŘÍLOHA P II: OBRÁZKY Z APLIKACE

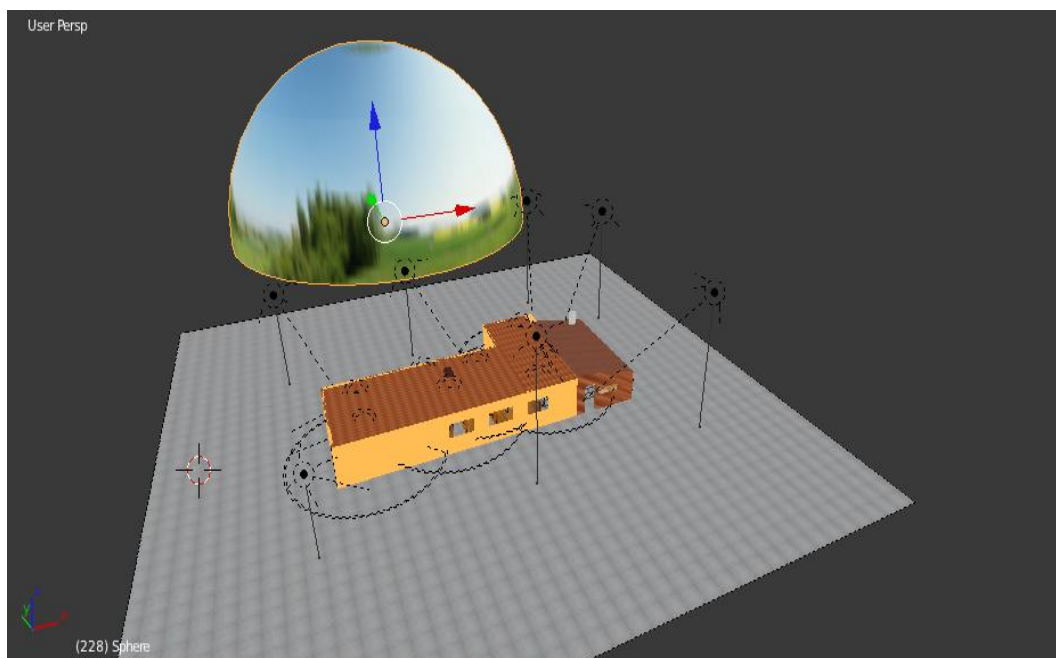


Obr. 32: Menu z aplikace

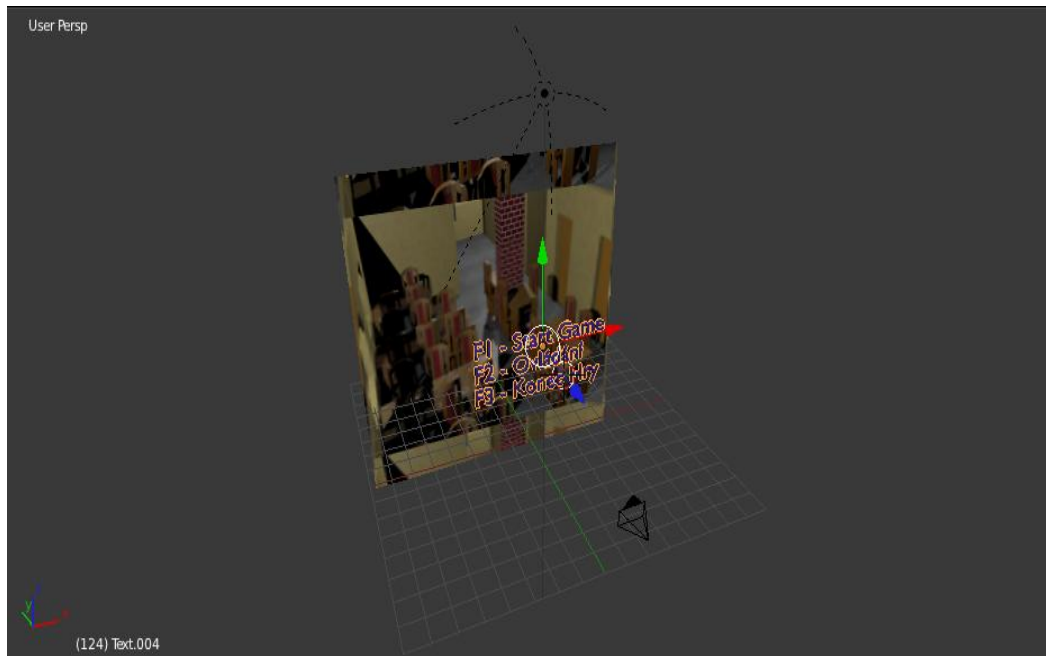


Obr. 33: Hra v aplikaci

PŘÍLOHA P III: OBRÁZKY SCÉN



Obr. 34: Scéna hospoda



Obr. 35: Scéna Menu