

Univerzita Hradec Králové
Přírodovědecká fakulta
Katedra informatiky

**Editace a modifikace map v enginech
id Tech 1 a id Tech 2**

Bakalářská práce

Autor: Pavel Tvrzník
Studijní program: B1801 Informatika
Studijní obor: Anglický jazyk se zaměřením na vzdělávání
Informatika se zaměřením na vzdělávání
Vedoucí práce: Mgr. Ondřej Kořínek

Hradec Králové

červenec 2016

Prohlášení

Prohlašuji, že jsem bakalářskou práci vypracoval samostatně a že jsem v seznamu použité literatury uvedl všechny prameny, které jsem při jejím psaní využil.

v Hradci Králové dne 13. 7. 2016

Pavel Tvrzník

Poděkování

Tímto bych chtěl poděkovat společnosti id Software a jejím zakladatelům za to, že vytvořili skvosty jako Wolfenstein 3D, Doom a Quake, bez nichž by herní průmysl nebyl tím, čím je dnes. Dále bych chtěl poděkovat všem, kdo mě při psaní této bakalářské práce podporovali, a stejně tak všem, kdo si práci přečetli.

Anotace

TVRZNÍK, Pavel. *Editace a modifikace map v enginech id Tech 1 a id Tech 2*. Hradec Králové, 2016. Bakalářská práce na Přírodovědecké fakultě Univerzity Hradec Králové. Vedoucí bakalářské práce Ondřej Kořínek. 71 s.

Tato bakalářská práce se zabývá počátky vývoje 3D herních engineů, přičemž důraz je kladen na enginey id Tech 1 a id Tech 2 a na hlavní představitele těchto engineů, Doom a Quake, pocházející od americké společnosti id Software. Práce je rozdělena na část teoretickou a část praktickou. Teoretická část práce vymezuje stěžejní pojmy vyskytující se v práci. Dále obsahuje přehled některých 3D herních engineů a historický průřez herními enginey od společnosti id Software a jejich význam ve vývoji 3D grafiky a herních engineů všeobecně. Následně uvádí a porovnává základní vlastnosti a možnosti dostupné v těchto enginech a na závěr představuje typy souborů a programy typické pro tyto enginey. Praktická část práce se podrobněji zabývá enginey id Tech 1 a id Tech 2, přičemž jejím výstupem je srovnání funkčních map v obou enginech vypracovaných specificky pro účely této práce.

Klíčová slova

Doom, engine, id Software, id Tech 1, id Tech 2, Quake

Annotation

TVRZNÍK, Pavel. *Editing and Modifications of Maps in id Tech 1 and id Tech 2 engines*. Hradec Králové, 2016. Bachelor Thesis at Faculty of Science University Hradec Králové. Thesis supervisor Ondřej Kořínek. 71 p.

This bachelor thesis deals with the beginnings of 3D game engines, with emphasis on id Tech 1 and id Tech 2 engines and on the lead exemplars of these engines, Doom and Quake, by the American company id Software. The thesis is divided into a theoretical part and a practical part. The theoretical part of the thesis defines the principal terms that appear in the thesis. It also contains an overview of some 3D game engines and a historical profile of id Software's game engines and their significance for the development of 3D graphics and game engines overall. After that it presents and compares the basic characteristics and possibilities available in these engines, and finally introduces the file types and programs typical for these engines. The practical part of the thesis deals more in depth with id Tech 1 and id Tech 2 engines, with its output being the comparison of the functional maps in both engines that have been made specifically for the purposes of this thesis.

Keywords

Doom, engine, id Software, id Tech 1, id Tech 2, Quake

Obsah

Úvod.....	8
1 Vymezení pojmů.....	10
1.1 Obecné pojmy	10
1.2 Pojmy specifické pro Doom engine	11
1.3 Pojmy specifické pro Quake engine	12
2 Herní enginy.....	13
2.1 Přehled některých FPS enginů	14
3 id Software a jeho působení na poli vývoje her a 3D grafiky	16
4 První 3D hry od id Softwaru	19
4.1 Wolfenstein 3D.....	19
4.2 Doom	22
4.3 Quake	24
5 Soubory a programy používané v Doom a Quake enginech.....	27
5.1 WAD.....	27
5.2 MAP, BSP a PAK.....	30
5.3 Doom Builder 2	31
5.4 TrenchBroom.....	32
6 Doom engine	34
6.1 Editace a modifikace	34
6.1.1 Geometrie map (vertex, linedef, sidedef, sector)	34
6.1.2 Parametry sektorů, úseček a stran úseček	37
6.1.3 Věci (things).....	45
6.1.4 Spuštěné události (tag, action).....	48
7 Quake engine	51
7.1 Editace a modifikace	51
7.1.1 Geometrie map (brush, face).....	51

7.1.2	Entity (entities)	54
7.1.3	Entity spuštěných událostí.....	57
7.1.4	Světla	59
8	Vlastní vytvořené mapy v Doom a Quake enginech	62
8.1	Pohled na mapy v editorech.....	62
8.2	Srovnání map	63
	Závěr.....	68
	Seznam použité literatury	69
	Seznam příloh	71

Úvod

Bakalářská práce se zabývá Editací a modifikací map v enginech id Tech 1 a id Tech 2. Jedná se o herní enginey, které americká společnost id Software vyvinula pro své stěžejní a nejproslulejší herní tituly: *Doom* a *Quake*, po nichž se příslušné enginey také původně nazývaly.

Práce je rozdělena na část teoretickou, která sestává z prvních pěti kapitol, a na část praktickou, která sestává ze zbylých třech kapitol.

Cílem teoretické části práce je za pomoci jak autorových vlastních znalostí, tak s užitím externích zdrojů, uvést problematiku herních engineů, provést jejich stručné historické vymezení a přehled některých jejich důležitějších vlastností. Dalším cílem teoretické části práce je přehled základních možností Doom a Quake engineu, které jsou potřebné k praktické části práce.

Cílem praktické části práce je na základě především autorových vlastních znalostí a zkušeností a jeho vlastní práce se soubory her *Doom* a *Quake* ukázat praktické využití pojmů nastíněných v teoretické části práce a porovnat tyto enginey pomocí vlastních vypracovaných map. Oběma výše uvedeným titulům, resp. jejich příslušným engineům, je vždy věnována jedna kapitola. V každé kapitole jsou ukázány možnosti tvorby map v daných enginech. Poslední kapitola praktické části porovnává vlastní mapy vypracované v těchto dvou enginech.

Obě části obsahují řadu obrázků vytvořených specificky pro potřeby této práce. Zpravidla se jedná o screenshoty pořízené v programech ZDoom a Doom Builder 2 v případě Doom engineu a v DirectQ a TrenchBroom v případě Quake engineu. Tyto obrázky slouží k výstižnější demonstraci problematiky, a stejně tak k jejímu snazšímu pochopení.

Kapitola první, *Vymezení pojmů*, vymezuje některé stěžejní pojmy, které budou uvedeny v obou částech práce, a jejichž znalost je zcela zásadní.

V kapitole druhé, která nese název *Herní enginey*, bude uveden pojem herní engine a jeho zásadní podíl na možnostech a funkčnosti her všeobecně. Kapitola dále bude obsahovat výběr z několika herních engineů a herních titulů, které daný engine užívají.

V kapitole třetí, která nese název *id Software a jeho působení na poli vývoje her a 3D grafiky*, bude stručně představena společnost id Software a její historie.

Kapitola čtvrtá nese název *První 3D hry od id Softwaru*. Kapitola obsahuje stručné představení her *Wolfenstein 3D*, *Doom* a *Quake* a přehled základních vlastností enginů těchto her, přičemž většina z těchto vlastností je pozorovatelná i z perspektivy hráče.

Kapitola pátá nese název *Soubory a programy používané v Doom a Quake enginech*. Tato kapitola bude pojednávat o souborech typu WAD, ve kterých je kromě samotných map uložena i grafika, zvuky a další nedílné součásti her v Doom enginu. Dále se kapitola bude zabývat soubory MAP a BSP, které jsou součástí Quake enginu. V této kapitole budou také představeny programy pro vytváření map v daných enginech.

Kapitola šestá nese název *Doom engine*. V této kapitole budou nejprve představeny základní vlastnosti a možnosti tohoto enginu. Poté se kapitola bude zabývat již samotnou editací map v tomto enginu.

Kapitola sedmá nese název *Quake engine*. Tato kapitola bude protějškem kapitoly předchozí a stejně tak v ní budou nejprve představeny základní vlastnosti a možnosti tohoto enginu a nakonec opět samotná editace.

Kapitola osmá nese název *Vlastní vytvořené mapy v Doom a Quake enginech*. Tato kapitola je porovnáním přiložených funkčních map v obou enginech, přičemž obě mapy byly vytvořeny autorem práce specificky pro její účely. Tyto mapy jsou obsažené v přílohách práce.

Soubor mapy v Doom enginu se jmenuje *bc_doom.wad* a byl vytvořen v editoru Doom Builder 2. Soubor mapy v Quake enginu se jmenuje *bc_quake.map* a byl vytvořen v editoru TrenchBroom a následně zkompileován do formátu BSP pomocí programu Tyr-Utills (v0.15). Soubor zkompileované mapy se jmenuje *bc_quake.bsp*.

1 Vymezení pojmů

V této kapitole je stručné vymezení některých důležitých pojmů, které se budou v této práci opakovaně používat. Kapitola je členěná do třech podkapitol, přičemž každá z nich je dedikována terminologii z příslušné oblasti.

Definice pojmů uvedené v **Podkapitole 1** byly zformulovány na základě všeobecných znalostí autora práce.

Ačkoli pojmy uvedené v **Podkapitolách 2 a 3** pravděpodobně nepatří mezi všeobecně známé informace, uvedené definice těchto pojmů byly taktéž zformulovány na základě vlastních znalostí a dlouhodobých zkušeností autora práce. Tyto pojmy budou podrobněji rozvedeny v praktické části práce v kapitolách zabývajících se příslušnými enginy. Vzhledem k absenci vystižného českého ekvivalentu pro některé z těchto pojmů zde budou uvedeny i pojmy, které v této práci budou dále užívány.

1.1 Obecné pojmy

id Tech 1 – Původně označován jako Doom engine. Název id Tech 1 vznikl při zavedení názvosloví id Tech při vydání enginu id Tech 5.

id Tech 2 – Původně označován jako Quake engine. Název id Tech 2 vznikl při zavedení názvosloví id Tech při vydání enginu id Tech 5.

Doom engine – Viz id Tech 1. V této práci bude užíváno zpravidla toto označení.

Quake engine – Viz id Tech 2. V této práci bude užíváno zpravidla toto označení.

FPS – Zkratka pro first-person shooter, česky střílečka z pohledu první osoby. Jedná se o žánr akčních her. Hry, kterými se tato práce zabývá, jsou právě tohoto žánru.

Singleplayer – Herní mód jednoho hráče. Je součástí většiny her.

Multiplayer – Herní mód více hráčů. V dnešní době probíhá téměř výhradně online.

Deathmatch – Varianta multiplayeru, kde více hráčů bojuje proti sobě.

Cooperative – Varianta multiplayeru, kde více hráčů spolupracuje, zpravidla proti počítači.

Secret – Tajný prostor typický pro *Doom*, *Quake* a jiné hry nejen žánru FPS.

Sprite – 2D obrázek používaný k zobrazování různých objektů ve hrách, které ještě nepodporovaly 3D modely. Pomocí spritů mohou být tvořeny animace u těchto objektů a stejně tak tyto objekty mohou být nasnímány z více různých úhlů.

1.2 Pojmy specifické pro Doom engine

WAD – Typ souboru. Jedná se o typ archivu, z něhož hry v Doom engine čtou většinu dat. WAD soubory obsahují mapy a informace o levelech, ale i veškeré grafické a zvukové soubory, stejně jako další data nezbytná pro běh hry. WAD je zkratkou pro "Where's All the Data?". [1, str. 134]

ZDoom – Source port pro hraní her v Doom engine. První verze vyšla v roce 1998, avšak nové verze vychází dodnes. Jedná se o freeware.

Doom Builder 2 – Editor pro tvorbu map v Doom engine. Tento editor vyšel v roce 2009. Jedná se o freeware.

XWE – Program pro manipulaci s WAD soubory. Jedná se o freeware.

Vertex – Jeden ze základních objektů tvořících geometrii map v Doom engine. Má podobu bodu, jedná se tedy o bezrozměrný rovinný geometrický útvar. Je určen dvěma souřadnicemi – X a Y. Českým ekvivalentem by byl "vrchol" nebo "uzel", avšak v této práci bude užíván termín "vertex".

Linedef – Jeden ze základních objektů tvořících geometrii map v Doom engine. Má podobu úsečky spojující dva vertexy. Linedef je zkratka pro "line definition", tedy definice úsečky. Linedef může mít buď jednu, nebo dvě strany. V této práci pro podobnost tohoto objektu s úsečkou a z důvodu snazšího skloňování tohoto termínu bude dále používán termín "úsečka".

Sidedef – Jeden ze základních objektů tvořících geometrii map v Doom engine. Jedná se o vlastnosti strany objektu úsečky. V této práci pro vlastnosti tohoto objektu a z důvodu snazšího skloňování tohoto termínu bude dále používán termín "strana úsečky".

Sector – Jeden ze základních objektů tvořících geometrii map v Doom engine. Má podobu mnohoúhelníku, jehož tvar je určen příslušnými úsečkami. V této práci bude dále užíván český ekvivalent "sektor".

Thing – Jeden ze základních objektů vyskytujících se v mapách v Doom engine. Tyto objekty reprezentují např. hráče, veškeré nepřátele, sebratelné předměty apod. V této práci bude dále užíván český ekvivalent "věc".

1.3 Pojmy specifické pro Quake engine

MAP – Typ souboru. Soubor MAP je samotný soubor, který tvůrce mapy při editaci mapy v Quake engine edituje. Jedná se o soubor obsahující prostý text, který nelze spustit. Z toho důvodu je nezbytné jej nejprve zkompilovat do formátu BSP pomocí softwaru k tomu určenému.

BSP – Typ souboru. Soubor BSP je souborem, který obsahuje zkompilovanou mapu pro Quake engine společně s texturami použitými v této mapě. Tento soubor již lze spustit a hrát. BSP je zkratkou pro Binary Space Partitioning. [2]

DirectQ – Source port pro hraní hry Quake. Jedná se o freeware.

TrenchBroom – 3D Editor pro tvorbu map v Quake engine. Jedná se o freeware.

Brush – Základní geometrický objekt tvořící mapy v Quake engine. Má podobu trojrozměrného geometrického útvaru. Brush reprezentuje pevnou oblast jako je např. podlaha, zeď, strop, ale i nebe apod. Jedním z možných českých ekvivalentů by byl např. "štětec", avšak pro nevýstižnou povahu tohoto českého výrazu a zároveň z důvodu snadného skloňování termínu "brush" bude tento termín v práci užíván i nadále.

Face – Rovina definovaná třemi body v trojrozměrném prostoru. Face pokaždé náleží nějakému brushi, a protnutí těchto rovin definuje výsledný tvar daného brushe. Termín "face" se používá i v geometrii pro stěny trojrozměrných geometrických objektů, takže v této práci bude užíván termín "stěna".

Entity – Entity v Quake engine reprezentují vše, co by v Doom engine bylo reprezentováno věcmi, tedy hráče, veškeré nepřátele, sebratelné předměty apod., avšak entity zde reprezentují i nestatické objekty jako dveře, výtahy apod. Speciálním případem entity je také entita "worldspawn".

Worldspawn – Specifická entita obsahující veškeré statické brushe. Je součástí každé mapy v Quake engine.

2 Herní enginey

Herní engine (anglicky motor, pohon) tvoří jádro většiny současných her. Tento pojem se začal postupně užívat v průběhu 90. let, kdy docházelo k rozvoji počítačové grafiky ve 3D, jejíž význam začal vzrůstat s nástupem žánru FPS. Díky herním engineům vývojáři her nemusí vyvíjet celou hru zcela od nuly tak, jak tomu bylo v počátcích herního průmyslu, kdy byly hry relativně jednoduché a odehrávaly se ryze ve 2D. [3]

Herní enginey jsou soustavami programů, které definují hranice, v jejichž rámci se vývojáři samotné hry mohou pohybovat. Jak bude detailněji uvedeno v dalších částech této práce, například Doom engine, na rozdíl od Quake engine a engineů novějších, neumožňuje použití dvou pater nad sebou, což je dáno dvojrozměrností geometrie mapy. Součástí engineů je zpravidla i renderovací engine, který obsahuje metodu renderování výsledného obrazu. Renderovacími enginey se však tato práce hlouběji zabývat nebude. [3]

Jak již bylo naznačeno, ne pro každou hru její tvůrci vyvíjí vlastní engine. V dnešní době je v herním průmyslu běžným jevem, že vývojáři používají engine od třetí strany a pouze ho upraví pro potřeby vlastního produktu, přidají svou grafiku, zvuky, 3D modely, herní logiku, chování apod. Například herní enginey z řady Unreal Engine od společnosti Epic Games tvoří jádra stovek více i méně známých herních titulů. [4] Je všeobecně známo, že se tak děje pochopitelně z finančních i časových důvodů, protože současné herní enginey jsou tvořeny velmi rozsáhlými a komplikovanými kódy.

Je zcela běžným jevem, že hry v současnosti užívají kromě svého jádrového engineu ještě několik dalších engineů specializovaných na konkrétní úkony ve hrách, které zpravidla přidávají na realističnosti. Přední společností ve vývoje těchto engineů je Havok, která je autorem engineů simulujících například fyzikální vlastnosti (Havok Physics), destrukci prostředí (Havok Destruction) nebo přirozené chování a animaci látky a vlasů (Havok Cloth). [5]

Modularita herních engineů také umožňuje jejich využití i v různých oblastech vědeckého výzkumu při 3D simulacích apod. Použitelnost herních engineů tedy není omezena pouze na herní průmysl. [6, str. 28 - 29]

Další často přehlíženou skutečností je, že rozvoj na poli herních enginů je hlavní motivací k vývoji nového hardwaru. [6, str. 27]

2.1 Přehled některých FPS enginů

Následující tabulka (**Tabulka 1**) slouží jako ilustrační chronologický přehled některých starších i novějších 3D herních enginů designovaných pro žánr FPS a herních titulů, jejichž jádro příslušný engine tvoří. Ačkoli uvedené tituly jsou převážně hry z pohledu první osoby, jsou mezi nimi i některé z pohledu osoby třetí. Ne všechny zde uvedené herní tituly jsou střílečky, jsou mezi nimi i adventury nebo RPG. U herních titulů jsou dále v závorce uvedeny roky jejich vydání. V žádném případě se nemá jednat o úplný seznam, jedná se pouze o přehled nejvýznamnějších enginů. Tabulka byla zpracována za použití článku z Wikipedie nesoucího název *List of first-person shooter engines*. [7]

Tabulka 1: Přehled herních enginů

Engine	První titul	Další významné tituly
Wolfenstein 3D engine	Wolfenstein 3D (1992)	Spear of Destiny (1992), Blake Stone: Aliens of Gold (1993), Rise of the Triad (1994), ...
id Tech 1 (Doom)	Doom (1993)	Doom II: Hell on Earth (1994), Heretic (1994), Hexen: Beyond Heretic (1995), ...
Build Engine	Witchaven (1995)	Duke Nukem 3D (1996), Redneck Rampage (1997), Blood (1997), ...
id Tech 2 (Quake)	Quake (1996)	Hexen II (1997), ...
id Tech 2 (Quake II)	Quake II (1997)	Heretic II (1998), Kingpin: Life of Crime (1999), Soldier of Fortune (2000), ...
Unreal Engine	Unreal (1998)	Unreal Tournament (1999), Deus Ex (2000), Clive Barker's Undying (2001), ...
GoldSrc	Half-Life (1998)	Team Fortress Classic (1999), Counter-Strike (2000), Day of Defeat (2003), ...

id Tech 3	Quake III Arena (1999)	Return to Castle Wolfenstein (2001), Medal of Honor: Allied Assault (2002), Soldier of Fortune II: Double Helix (2002), Call of Duty (2003), ...
LS3D	Mafia (2002)	Hidden & Dangerous (2004), ...
Unreal Engine 2.0	America's Army (2002)	Unreal Tournament 2003 (2002), Unreal II: The Awakening (2003), Tom Clancy's Rainbow Six 3: Raven Shield (2003), Lineage II (2003), ...
PainEngine	Painkiller (2004)	Painkiller: Overdose (2007), NecroVisioN (2009), ...
Unreal Engine 2.5	Tribes: Vengeance (2004)	Unreal Tournament 2004 (2004), BioShock (2007), Duke Nukem Forever (2011), ...
CryEngine	Far Cry (2004)	Aion: Tower of Eternity (2008), ...
id Tech 4	Doom 3 (2004)	Quake 4 (2005), Prey (2006), Wolfenstein (2009), ...
Source	Half-Life 2 (2004)	Counter-Strike: Source (2004), Team Fortress 2 (2007), Portal (2007), Portal 2 (2011), Counter-Strike: Global Offensive (2012), ...
Unreal Engine 3	Gears of War (2006)	Tom Clancy's Rainbow Six: Vegas (2006), Unreal Tournament 3 (2007), Mirror's Edge (2008), Borderlands (2009), Tera (2011), BioShock Infinite (2013), ...
Frostbite	Battlefield: Bad Company (2008)	Battlefield: Bad Company 2 (2010), Medal of Honor multiplayer (2010), ...
id Tech 5	Rage (2011)	Wolfenstein: The New Order (2014), The Evil Within (2014), ...
id Tech 6	Doom (2016)	

3 id Software a jeho působení na poli vývoje her a 3D grafiky

id Software založili v roce 1991 programátoři John Carmack a John Romero, designér Tom Hall a grafik Adrian Carmack. Tato čtveřice dříve působila ve společnosti Softdisk. [8]

Zakladatelé společnosti id Software ještě během svého působení v Softdisku vyvinula hru *Commander Keen*, která užívala *Adaptive Tile Refresh*, metodu pro plynulé scrollování ve 2D vyvinutou Johnem Carmackem. [1, str. 42] Poté, co založili id Software, ještě vázání smlouvou se Softdiskem, vytvořili několik herních titulů publikovaných svou bývalou společností, mezi něž patřil i *Hovortank 3D*. Engine užíval renderovací metodu *raycasting*¹, kterou navrhl opět John Carmack, díky čemuž je považován za přelomový počín v zobrazování 3D grafiky z pohledu první osoby. Svým způsobem takto došlo ke vzniku nového žánru her – FPS. [1, str. 68] Dalším titulem hrajícím roli ve vývoji 3D enginů byl *Catacomb 3D*. Zatímco v *Hovortanku 3D* byly všechny stěny jednobarevné, engine *Catacomb 3D* již umožňoval vykreslování textur. [1, str. 75]

Po *Catacomb 3D* přišel přelomový titul id Softwaru. 5. května 1992 byl vydán *Wolfenstein 3D*, hra, kterou se id Software zapsal do historie herního průmyslu a která započala obrovský nárůst popularity her žánru FPS.

Po obrovském úspěchu *Wolfensteinu 3D* id Software začal pracovat na jeho prequelu, *Spear of Destiny*, který však užívá identický engine jako *Wolfenstein 3D* a obsahuje pouze některé nové textury, nepřátele a objekty. Zatímco většina id Softwaru pracovala na *Spear of Destiny*, John Carmack vyvíjel nový 3D engine, který měl v mnohém *Wolfenstein 3D* engine předčít. Jednalo se o Doom engine, dnes oficiálně nazývaný id Tech 1. [1, str. 98 - 101]

Doom byl vydán 10. prosince 1993 a dosáhl ještě většího úspěchu než jeho předchůdce. Doom engine prorazil s revolučními možnostmi: zdi již nemusely být vzájemně kolmé, v mapách existovaly výškové rozdíly, textury se aplikovaly i na

¹ Metoda, jejímž účelem je urychlení renderování výsledného obrazu a šetření operační paměti. Toho je docíleno pomocí paprsků vycházejících z pozice hráče, které detekují kolizi s objekty, podle čehož engine pozná, že nemá renderovat objekty nacházející se za stěnami. [1, str. 68]

podlahy a stropy, existovaly zde světelné rozdíly a různé spouštěné události, které umožňovaly otevírání dveří na dálku, výtahy, teleporty, vysouvání plošin a řadu dalších věcí, které hře přidávaly na dynamičnosti. Další revoluční součástí *Doomu* byl multiplayer, který byl v žánru FPS bezprecedentní. Hraní multiplayeru bylo možné buď online (tehdy přes telefonické připojení přes modem), nebo offline za propojení více počítačů pomocí null-modem kabelu. [1, str. 120-122]

Mimo možností, které tvůrcům map umožňoval, byl *Doom* skokem kupředu i z grafického hlediska. Zajímavostí je, že kvůli nedostatečnému výkonu tehdejšího hardwaru bylo rozlišení v *Doomu* limitováno na 320 x 200 a 16bitovou barevnou hloubku, protože podle Johna Carmacka byla vyšší rozlišení s vyšší barevnou hloubkou vzhledem k dostupnému hardwaru zbytečná. [9]

Pokračování *Doomu* s názvem *Doom II: Hell on Earth* vyšlo 30. září 1994. *Doom II* běžel na stejném enginu jako první díl, opět byly pouze přidány nové textury, noví nepřátelé, nová zbraň a nové objekty. Vztah mezi *Doomem II* a *Doomem* lze přirovnat ke vztahu *Spear of Destiny* a *Wolfensteinu 3D*.

K dalšímu obrovskému skoku kupředu došlo se vznikem Quake enginu, dnes nazývaného id Tech 2. *Quake* vyšel 22. června 1996 a jeho engine již podporoval skutečný 3D design map. Dále se zde objevila řada dnes zcela běžných součástí 3D her: engine využíval real-timeový 3D rendering a podporu pro hardwarovou akceleraci při 3D renderování prostřednictvím OpenGL. Engine dále využíval polygonální 3D modely a v neposlední řadě lightmapping, který v případě *Quaku* byl zatím pouze předem renderovaný. [10]

Ačkoli se tato práce dále bude zabývat hlavně *Doom* a *Quake* enginy, je třeba nejprve zmínit několik dalších enginů a titulů z dílny id Softwaru, které po nich následovaly.

Pokračovatelem Quake enginu použitého v titulech *Quake* a *Quake II* byl Quake III Arena engine, dnes nazývaný id Tech 3. Jak už původní název napovídá, tento engine byl vytvořen pro titul *Quake III Arena* a opět oproti svému předchůdci obsahoval několik vylepšení. Na rozdíl od svých předchůdců již ani neobsahoval softwarový rendering a ke svému chodu vyžadoval grafický procesor s podporou pro OpenGL. Engine dále umožňoval použití zakřivených povrchů, podporoval pokročilé shadery a dynamická světla. [11]

Příštím enginem z této řady byl *Doom 3* engine, dnes nazývaný *id Tech 4*. Oproti *Quake III Arena* enginu obsahoval různorodé pokročilé techniky texturování, jako je např. *bump mapping*, a pokročilejší shadery. Stíny a světla v *Doom 3* enginu jsou plně dynamické a jejich výpočet při renderování probíhá pro každý pixel zvlášť. Dále zde byla uvedena metoda *specular highlighting*, která na určité místo na osvětleném objektu přidává světlý odlesk, který má nemalý podíl na jeho realistickém vzhledu. [12]

Následujícím enginem byl *id Tech 5*, který byl již poprvé licencován pod tímto názvem. Při jeho vydání byly výše zmíněné enginey z dílny id Softwaru retrospektivně pojmenovány, aby zapadaly do této nomenklatury. *id Tech 5* a jeho reprezentující hra *Rage* užívají techniky *MegaTexture 2.0*, která umožňuje podporu textur s extrémně vysokým rozlišením (až 128 000 x 128 000), což dodává na realističnosti hlavně rozsáhlým jasně osvětleným venkovním prostorám (pro srovnání, textury používané v *Doom* a *Quake* enginech měly typicky rozlišení 64 x 64, takže jejich opakování při aplikaci na model bylo jasně viditelné). Další vlastností tohoto enginu je renderování polostínu na hranách prostředí a stínů. Tyto přechody jsou v tomto enginu oproti jeho předchůdci jemné a opět realističtější. [13]

Nejaktuálnějším enginem z řady id Tech je *id Tech 6*. Na tomto enginu staví titul *Doom*, který byl vydán 13. května 2016 (původně *Doom 4*, který měl používat *id Tech 5*).

4 První 3D hry od id Softwaru

V této kapitole budou popsány první tři stěžejní tituly vytvořené společností id Software: *Wolfenstein 3D*, *Doom* a *Quake*. Všechny z těchto tří her byly podstatnými milníky ve vývoji 3D her a 3D grafiky všeobecně, a možnosti, kterými tyto enginy disponují, budou v této kapitole představeny.

Každá ze tří následujících podkapitol bude obsahovat dva obrázky, přičemž jeden z nich zachycuje prostředí hry a druhý mapu levelu zachyceného v obrázku prvním. Poté vždy následuje několik bodů popisujících základní vlastnosti enginu, které jsou pozorovatelné i z perspektivy běžného hráče.

Tato kapitola bude pouze teoretická, přičemž tento popis je stěžejní pro porovnání v praktické části práce.

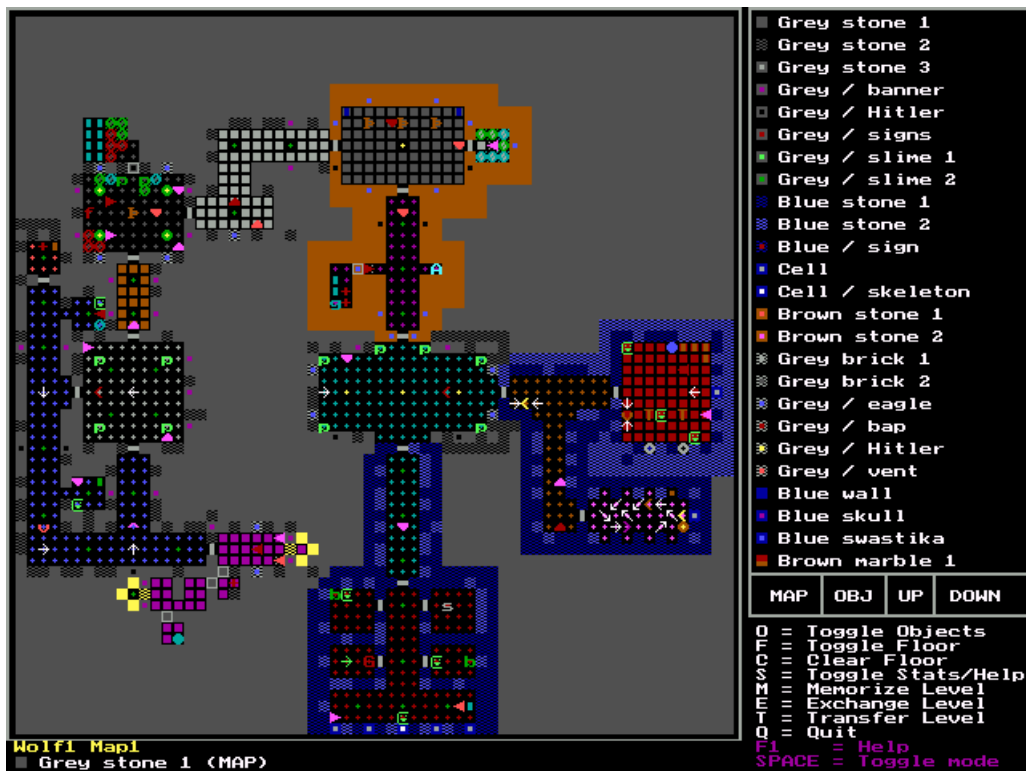
4.1 Wolfenstein 3D

Obrázek 1 zachycuje jednu z charakteristických místností z *E1L1* (1. levelu 1. epizody; obrázek z důvodů kompatibility nepochází přímo z původního *wolf3d.exe* pro MS-DOS, ale jedná se o *Wolfenstein 3D TC* pro ZDoom). Na tomto obrázku lze pozorovat některé z klíčových vlastností tohoto enginu.

Obrázek 2 je mapou *E1L1* z editoru pro *Wolfenstein 3D* pro MS-DOS. Místnost zachycená na **Obrázku 1** je místnost uprostřed mapy, přičemž hráč je natočen na severovýchod.



Obrázek 1: Wolfenstein 3D E1L1 - hra



Obrázek 2: Wolfenstein 3D E1L1 - mapa (zdroj: <http://www.belowe.com/images/LostEp/w11.gif>)

Následuje přehled základních vlastností Wolfenstein 3D enginu, přičemž některé z nich jsou pozorovatelné na **Obrázcích 1 a 2**:

1. V levelech nejsou žádné výškové rozdíly, takže vše je neustále ve stejné úrovni.
2. Všechny podlahy a stropy mají stejnou barvu. V případě podlah to je vždy světle šedá, v případě stropů barva závisí na levelu.
3. Jak je patrné z **Obrázku 2**, všechny stěny v levelu jsou vzájemně buď kolmé, nebo rovnoběžné. To je dáno tím, že mapy mají formát čtvercové sítě o rozměrech 64 x 64. Zdi se na mapu umísťují jako krychlové bloky, přičemž na jednu mapovou jednotku může připadat pouze jeden blok zdi nebo jiný objekt. Důsledkem tohoto formátu map je, že žádné různoběžné stěny ve Wolfenstein 3D enginu neexistují.
4. V prostředí levelů jsou interaktivními prostředky pouze dveře, které se otevírají do strany, tajné zdi, které se posunou o 3 mapové jednotky a v případě, že je trajektorie jejich posunu blokována jinou zdí nebo statickým objektem, se zastaví, a tlačítko výtahu, kterým se ukončí level.
5. Veškeré statické objekty (lustry, lampy, stoly apod.) a sebratelné předměty (náboje, lékárny, poklady apod.) jsou tvořeny jedním předem renderovaným 2D spritem, čehož důsledkem je, že se objekty vždy natáčejí vzhledem k hráči. To je patrné zvláště u mrtvol. Pro představu, i kdyby hráč na **Obrázku 1** stál u modré zdi úplně vpravo a byl natočen směrem k hnědé zdi viditelné za živým nepřítelem vlevo, mrtvola v pravé části obrázku by kromě vzdálenosti vypadala naprosto identicky.
6. Nepřátelé nejsou tvořeni 3D modely, nýbrž několika předem renderovanými 2D sprity. Tyto sprity zachycují danou postavu při různých úkonech – při stání, při chůzi, při útoku a při umírání. Sprity pro stání a chůzi jsou navíc nasnímány z osmi různých úhlů (s výjimkou bossů, kteří jsou vždy otočeni čelem k hráči). Ostatní sprity mají pouze jeden úhel. Po smrti nepřítele se jeho mrtvola stane statickou.

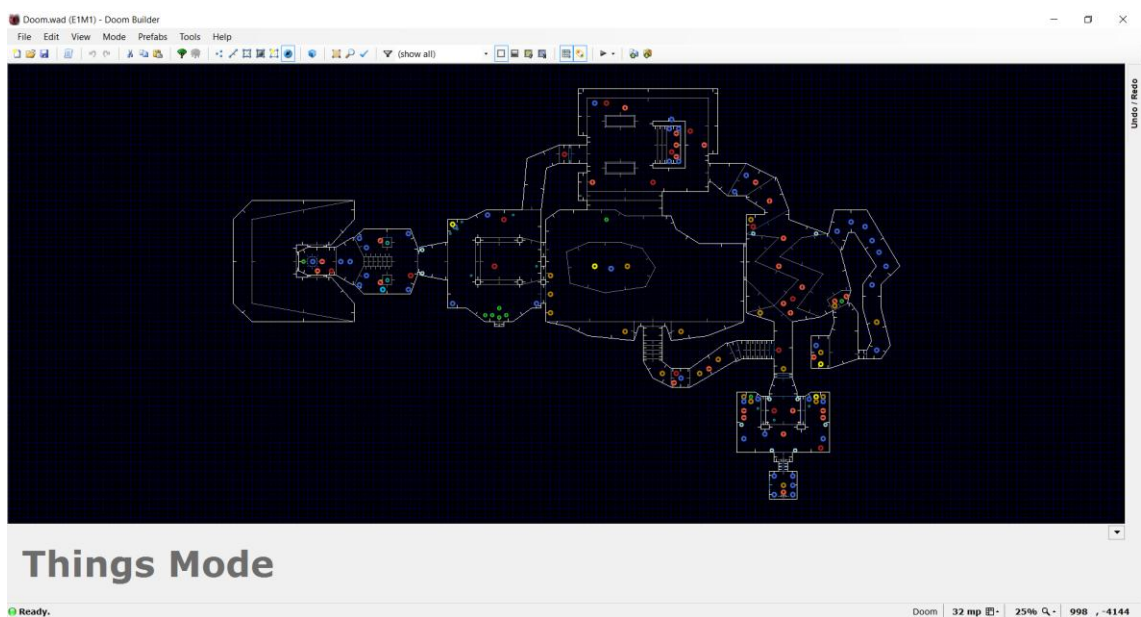
4.2 Doom

Obrázek 3 zachycuje první místnost z *E1M1: Hangar* (1. mapy 1. epizody; obrázek byl pořízen v programu ZDoom verze 2.3.1). Na tomto obrázku lze pozorovat některé klíčové vlastnosti tohoto engine.

Obrázek 4 je mapou téže mapy a byl pořízen v editoru Doom Builder 2 verze 2.1.2. Místnost zachycená na **Obrázku 3** je místnost zhruba uprostřed mapy a hráč je opět natočen na severovýchod.



Obrázek 3: Doom E1M1 – hra



Obrázek 4: Doom E1M1 – mapa otevřená v editoru Doom Builder 2

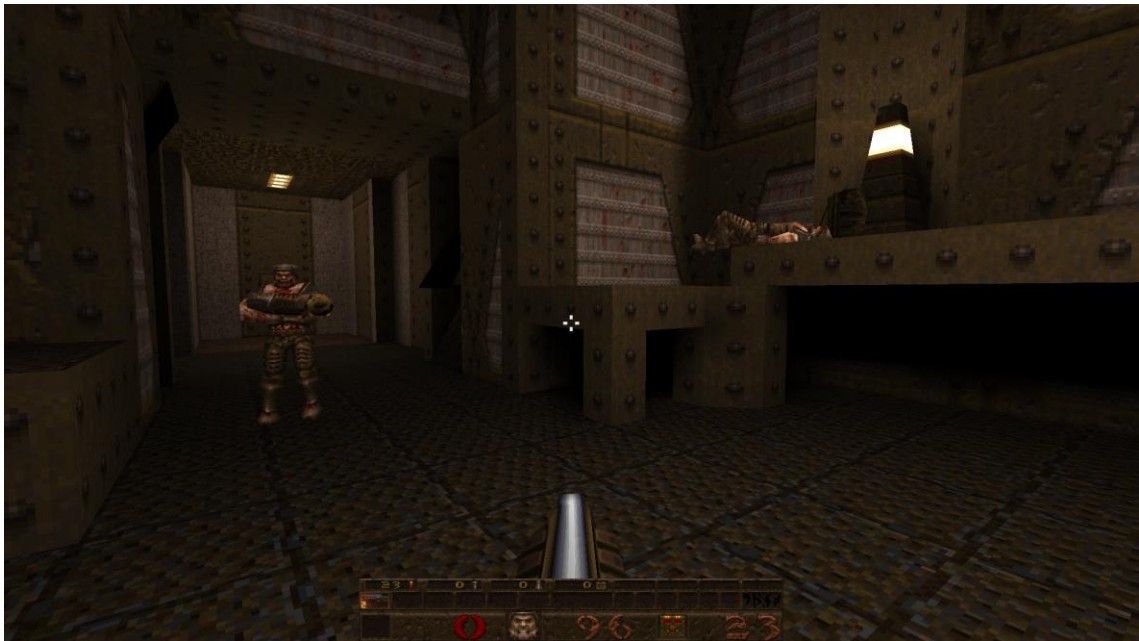
Následuje přehled základních vlastností Doom engine, přičemž některé z nich jsou opět pozorovatelné na **Obrázcích 3 a 4**:

1. V levelech jsou výškové rozdíly – jak podlahy, tak stropy mají různou výšku.
2. Podlahy a stropy mají různé textury.
3. Jak lze vidět na **Obrázku 3**, zdi již nemusí být vzájemně kolmé. Prostory jsou tvořeny jako plošinné geometrické útvary.
4. Z **Obrázku 3** je také patrné, že zde existují světelné rozdíly. Světla mohou také blikat nebo pulzovat. Toto osvětlení je však vždy fixně nastaveno danému sektoru.
5. Předměty mohou být rozmístovány s výrazně větší volností než ve Wolfenstein 3D engine. Na stejných souřadnicích může být libovolný počet předmětů.
6. V prostředí levelů je mnohem více interaktivních objektů jako jsou různé typy dveří, výtahy, teleporty, výsuvné plošiny, drtiče apod. Většinu z nich lze spouštět pomocí spínačů nebo přejitím hráče přes určitou úsečku a některé z nich také výstřelem do spínače.
7. Podobně jako ve Wolfenstein 3D engine jsou veškeré statické objekty (lampy, stromy, visící těla apod.) a sebratelné předměty (náboje, lékárny, powerupy apod.) tvořeny jedním předem renderovaným 2D spritem, takže se opět natáčí vzhledem k hráči. U mrtvol to platí taktéž.
8. Nepřátelé opět nejsou tvořeni 3D modely, nýbrž několika předem renderovanými 2D sprity. Oproti *Wolfensteinu 3D* tu je však několik rozdílů. Sprity opět zachycují danou postavu při různých úkonech, avšak někteří nepřátelé, kteří jsou souměrní podle vertikální osy (např. Cacodemon), nejsou při stání a pohybu nasnímáni z osmi úhlů, ale pouze z pěti. Zbylé tři úhly jsou získány převrácením opačných spritů podle vertikální osy. Nesouměrní nepřátelé (např. Cyberdemon) jsou nasnímáni ze všech osmi úhlů. Sprity útočících nepřátel jsou nasnímány také z pěti nebo osmi úhlů, protože na rozdíl od *Wolfensteinu 3D*, kde se nepřátelé při útoku vždy otočili na hráče, zde mohou bojovat i mezi sebou. Sprity pro umírání a mrtvoly už mají pouze jeden úhel.

4.3 Quake

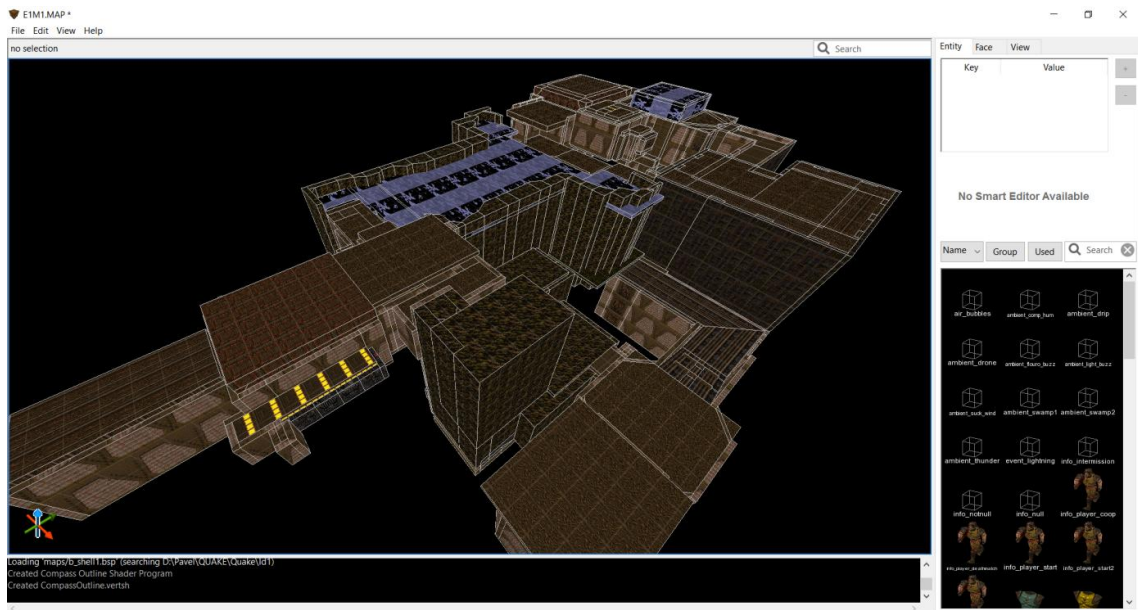
Obrázek 5 zachycuje první místnost z *E1M1: The Slipgate Complex* (1. mapy 1. epizody; obrázek byl pořízen v programu DirectQ verze 1.9.0). Na obrázku lze opět pozorovat řadu charakteristických vlastností enginu.

Obrázek 6 je pohledem na zdrojovou mapu *E1M1* v editoru TrenchBroom verze 1.1.6. Na první pohled lze zpozorovat, že mapa je plně trojrozměrná. Většina editorů obsahuje i 2D pohled, ten je však při oddálení a při absenci dokonalých znalostí mapy značně nepřehledný. **Obrázek 6** zachycuje pouze vnější kostru mapy v tzv. voidu², a tudíž na něm nelze pozorovat žádné interiéry.



Obrázek 5: Quake E1M1 – hra

² Česky "prázdnota". Tento termín se ve hrách běžně používá pro prázdný prostor obklopující samotnou mapu.



Obrázek 6: Quake E1M1 – mapa otevřená v editoru TrenchBroom

Nyní opět následuje přehled vlastností, které Quake engine získal oproti svému předchůdci. Některé z nich jsou opět pozorovatelné na **Obrázcích 5 a 6**:

1. Levely jsou plně trojrozměrné, takže nad sebou může být více pater, mohou zde být plošiny s volnými prostory nad sebou i pod sebou, různé mosty, mříže, zábradlí apod.
2. Levely mohou obsahovat šikmé plochy (na **Obrázku 5** je taková plocha viditelná například na lampě v pravé horní části obrázku).
3. Osvětlení v levelech už není fixně nastavené danému sektoru (Quake engine ani sektory jako Doom engine neuvžívá), ale funguje na principu světelných zdrojů, které světlo vyzařují a to se s rostoucí vzdáleností postupně vytrácí. Stejně tak se za překážkami vytvoří stín (to lze pozorovat v pravé části **Obrázku 5**, kde je pod plošinou stín).
4. V levelech může být voda, kyselina, láva a další kapaliny, do kterých se dá ponořit a plavat (v Doom engine byly kapaliny pouze běžnými animovanými texturami).
5. V levelech mohou být nejrůznější interaktivní objekty, které v Doom engine nebylo možné vytvořit. Je tu možnost horizontálního i vertikálního pohybu 3D objektů, díky čemuž lze zkonstruovat dveře, které se otevírají do strany, různé výsuvné mosty a plošiny, tajné zdi, které se zasunou a poté uhnou do strany apod. Engine dále umožňuje vytvářet plošiny a jiné objekty, které

jezdí dokola po určité dráze. V rozšíření *Scourge of Armagon* ještě přibyla možnost horizontální i vertikální rotace objektů.

6. Jeden spouštěč může spustit více různých událostí najednou. Např. stisknutím jednoho spínače lze současně otevřít dveře a vysunout most. Tyto události mohou proběhnout současně i postupně.
7. Mohou zde být různé sekvence, které vyžadují stisknutí více spínačů nebo zabití více nepřátel k tomu, aby došlo ke spuštění dané události.
8. Veškeré statické objekty, sebratelné předměty i nepřátelé jsou již tvořeni skutečnými 3D modely.

5 Soubory a programy používané v Doom a Quake enginech

V této kapitole budou popsány některé typy souborů typické pro Doom engine a Quake engine: soubory typu WAD pro Doom engine a soubory typu MAP, BSP a PAK pro Quake engine.

Poté budou představeny editory map pro oba enginey: Doom Builder 2 pro Doom engine a TrenchBroom pro Quake engine.

Kapitola obsahuje obrázky pořízené v programech XWE (eXtensible Wad Editor, verze 1.16), Doom Builder 2 a TrenchBroom.

5.1 WAD

WAD je zkratkou pro "Where's All the Data?". [1, str. 134] Jak název napovídá, jedná se o soubory obsahující podstatnou část hry, včetně grafiky, zvuků, a na neposledním místě samotných map.

WAD soubory jsou formou archivu s danou vnitřní strukturou, díky níž lze mít všechna data na jednom místě, odkud je může engine snadno číst.

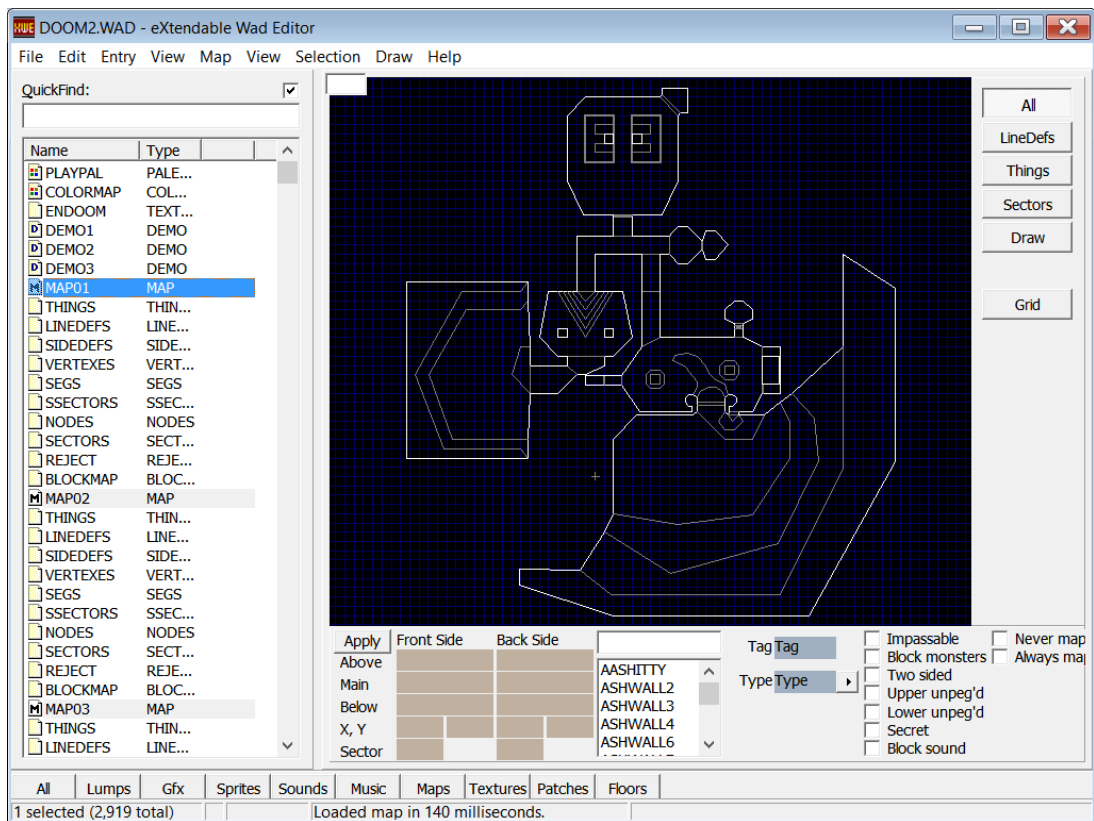
WAD soubory se pro potřeby Doom engineu dělí na dva typy: **IWAD** (internal WAD) a **PWAD** (patch WAD). [14]

IWAD je hlavním WAD souborem hry a k jeho spuštění již žádný další WAD soubor není potřeba. Obsahuje původní mapy, grafiku, zvuky a další soubory.

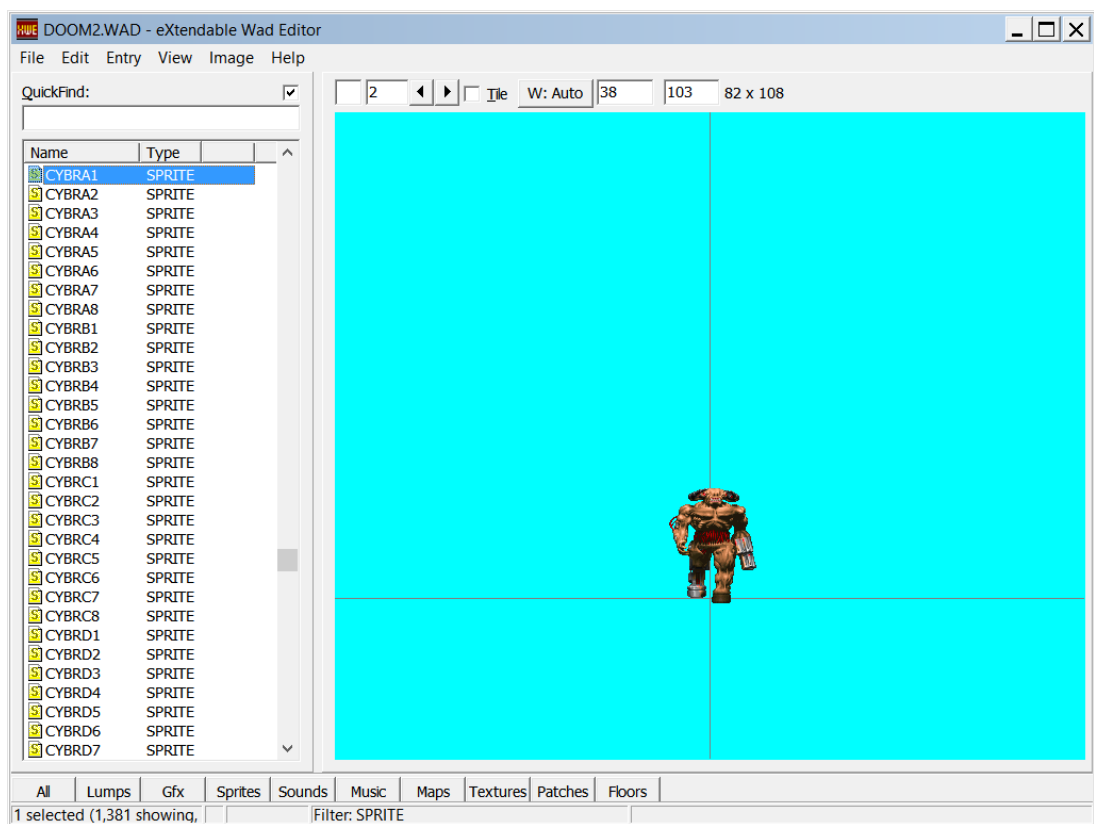
Nejvýznamnějšími IWAD soubory jsou *doom.wad* a *doom2.wad*, tedy WAD soubory titulů *Doom* a *Doom II: Hell on Earth* vydaných v letech 1993 a 1994. Dalšími podstatnými Doom IWADy jsou *tnt.wad* (*Final Doom – TNT: Evilution*) a *plutonia.wad* (*Final Doom – The Plutonia Experiment*). Ostatní hry běžící na tomto engineu také mají svůj IWAD – např. *heretic.wad* (*Heretic*) nebo *hexen.wad* (*Hexen: Beyond Heretic*).

PWAD je WAD soubor, jehož hlavním obsahem jsou mapy, které při jeho spuštění nahradí mapy v IWADu. PWAD samozřejmě může obsahovat i přídatné textury a grafiku, hudbu nebo zvuky, kterými se při spuštění PWADu opět nahradí ty původní z IWADu.

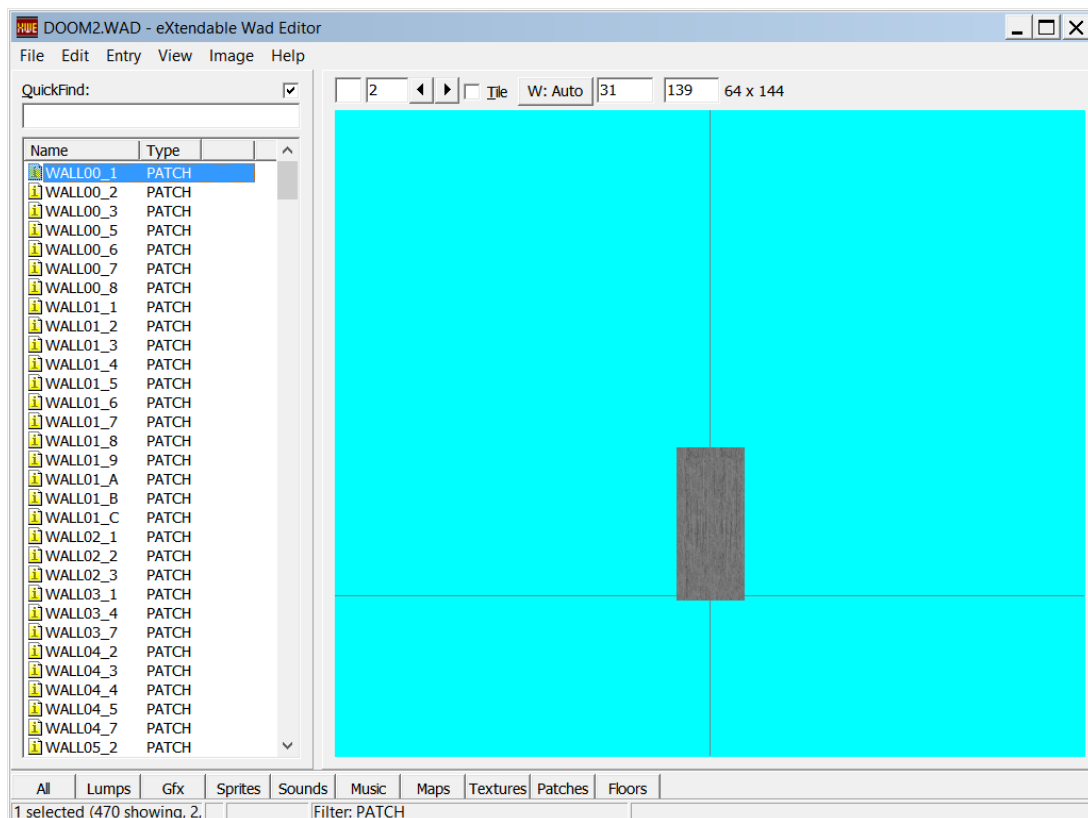
Následuje několik obrázků zachycujících prostředí programu XWE:



Obrázek 7: DOOM2.WAD otevřený v programu XWE 1



Obrázek 8: DOOM2.WAD otevřený v programu XWE 2



Obrázek 9: DOOM2.WAD otevřený v programu XWE 3

Na **Obrázcích 7, 8 a 9** lze vidět některé soubory obsažené v archivu DOOM2.wad otevřeném v programu XWE.

Na **Obrázku 7** je označená položka typu MAP jménem MAP01. K této položce náleží několik dalších položek, které jsou v archivu umístěny přímo pod ní: THINGS, LINEDEFS, SIDEDEFS, VERTEXES, SEGS, SSECTORS, NODES, SECTORS, REJECT a BLOCKMAP. Tyto položky jsou v podstatě tabulky obsahující hodnoty typu integer a string, které definují dané objekty v mapě. Více o nich bude v **Kapitole 6**. Tyto položky společně obsahují veškerá data z 1. levelu *Doom II: Hell on Earth, Entryway*.

Na **Obrázku 8** je označená položka typu SPRITE jménem CYBRA1. Jedná se o jeden z 68 spritů monstra jménem Cyberdemon.

Na **Obrázku 9** je označená položka typu PATCH jménem WALL00_1. Tento grafický soubor je použit ve třech texturách: GRAYBIG, GRAYTALL a GRAYVINE.

5.2 MAP, BSP a PAK

Zatímco pro editaci map pro Doom engine jsou typické pouze WAD soubory, v případě Quake engine je těchto specifických souborů více. V této práci budou zmíněny tři: MAP, BSP a PAK.

MAP je typem souboru spjatým s Quake engine, kterým se práce v **Kapitole 7** bude zabývat nejvíce. Je to soubor, který tvůrce mapy edituje. Na rozdíl od WAD souborů v Doom engine, kde se editací měnilo hned několik příslušných položek náležících editované mapě, v Quake engine to je pouze tento soubor. Opět je příhodné zmínit soubory typu WAD, protože z těchto souborů editor načítá textury používané při tvorbě MAP souboru. Další podstatnou skutečností je, že MAP soubor je pouze určitým způsobem formátovaný text. Aby mapa šla spustit, je jí nutné nejprve zkompileovat do souboru typu BSP pomocí specializovaného softwaru. [15]

BSP je zkratkou pro Binary Space Partitioning, což je metoda rozdělování prostoru do konvexních množin za pomoci nadrovin [2]. Takto vznikají tzv. BSP stromy, které kromě systematického dělení 3D prostoru umožňují i hledání v něm, čímž se značně urychlí renderování výsledného obrazu. [16, str. 1] Ačkoli tato metoda byla užita již v Doom engine, teprve v případě Quake engine po ní byl pojmenován typ souboru. BSP soubor již lze spustit a hrát. Do tohoto souboru se při kompilaci také ukládají textury, které v daném MAP souboru byly užity.

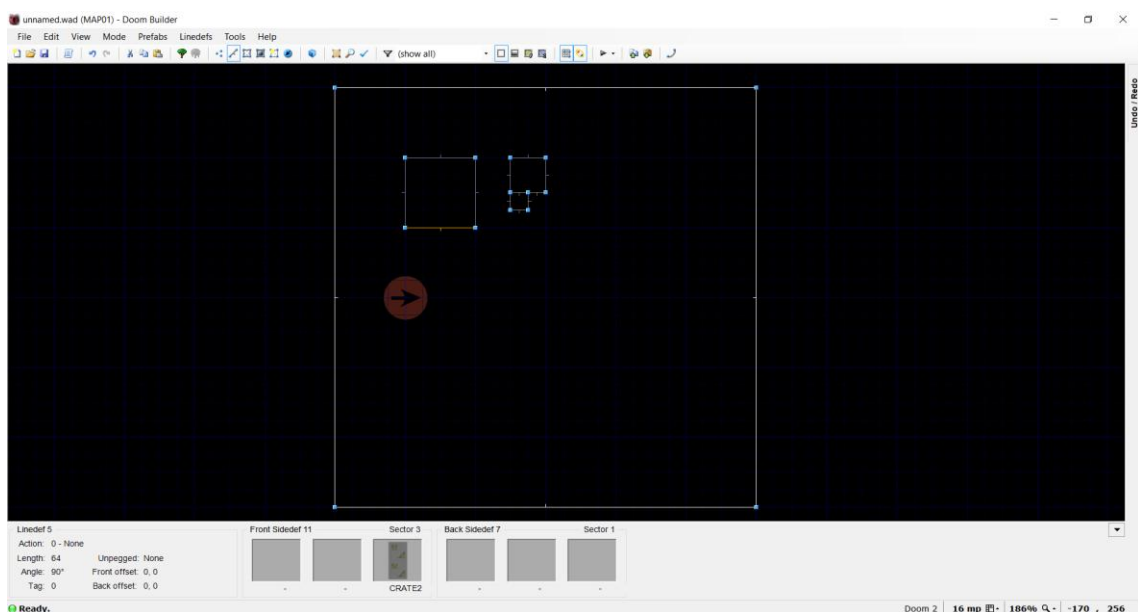
Protějškem WAD souboru je v případě Quake engine soubor PAK. Jedná se o modernější metodu komprese, která byla a stále je užívána některými herními enginy. PAK soubory obsahují mj. zvuky, modely a samozřejmě samotné mapy ve formátu BSP. K manipulaci s těmito soubory je opět potřeba specializovaný software, např. PakExplorer.

5.3 Doom Builder 2

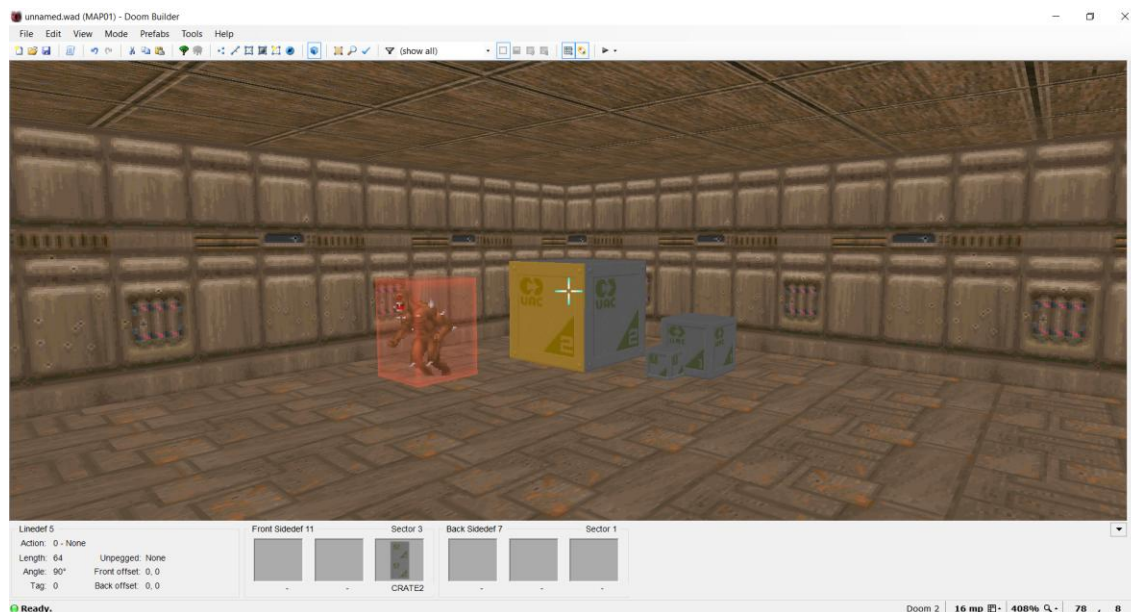
Doom Builder 2 je freeware, který vyvinul Pascal van der Heiden, v komunitě známý pod přezdívkou CodeImp. Byl vydán v květnu 2009. Jedná se o velice kvalitní editor pro editaci map pro řadu her stavějících na Doom engine. Doom Builder 2 má kvalitní interface s některými pokročilými funkcemi včetně různých možnosti zobrazování sektorů a dalších objektů, poměrně intuitivní ovládání a velice rozsáhlé možnosti nastavení včetně řady klávesových zkratk, které umožňují tvůrci maximální efektivitu při editaci. Mnohými členy komunity je Doom Builder 2 považován ze nejlepší Doom editor vůbec. [17]

Stejně jako v ostatních editorech pro mapy v tomto engine se pracuje primárně ve 2D prostředí, avšak Doom Builder 2 (stejně jako jeho předchůdce, Doom Builder) obsahuje i kvalitní 3D mód. Ten sice neumožňuje vytváření nebo posouvání geometrie map, ale umožňuje nastavování vlastností u různých objektů. Navíc poskytuje tvůrci lepší představu o autentickém vzhledu vytvářené mapy bez nutnosti mapu kompilovat a spouštět.

Obrázky 10 a 11 byly pořízeny v Doom Builderu 2 a zachycují jednoduchou čtvercovou místnost o rozměrech 384 x 384 mapových jednotek. **Obrázek 10** je zobrazením této místnosti ve 2D módu, zatímco **Obrázek 11** ve 3D módu.



Obrázek 10: Doom Builder 2 – Náhled na jednoduchou místnost ve 2D módu

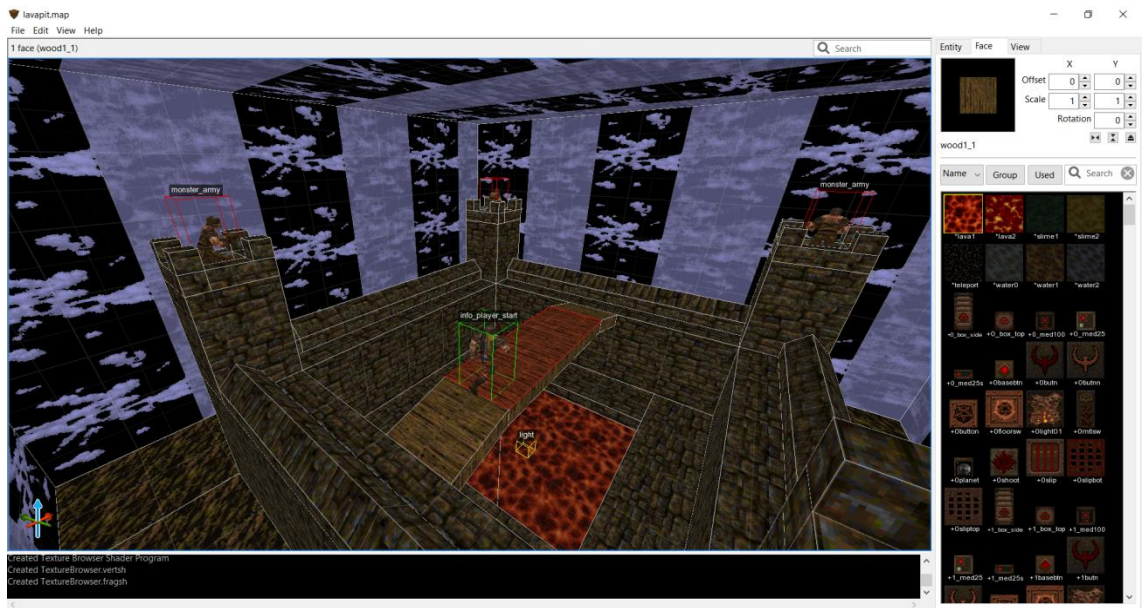


Obrázek 11: Doom Builder 2 – Náhled na jednoduchou místnost ve 3D módu

5.4 TrenchBroom

TrenchBroom je freeware, který vyvinul Kristian Duske. Jedná se o moderní a poměrně efektivní editor pracující plně ve 3D. Z toho důvodu je pro tvůrce map zásadní znalost a schopnost užívání klávesových zkratek. Editor obsahuje funkce pro efektivní manipulaci s vertexy, rozdělování hran brushů, horizontální i vertikální rotaci a převracení výběru a nástroj na ořezávání brushů pomocí roviny určené dvěma či třemi body. [18]

Obrázek 12 byl pořízen v editoru TrenchBroom a zachycuje místnost obsahující některé z možností, které má Quake engine oproti Doom engine navíc: 3D modely, více pater nad sebou, nakloněné roviny a kapaliny, do nichž se dá ponořit. Na obrázku lze také vidět entitu typu Light, která po kompilaci v softwaru na kompilaci lightmapů funguje jako zdroj světla.



Obrázek 12: TrenchBroom - Náhled na místnost obsahující prvky Quake enginu

6 Doom engine

V **Kapitole 4** byly mimo jiné uvedeny základní vlastnosti a možnosti Doom engineu a v **Kapitole 5** byly představeny WAD soubory a program Doom builder 2, ve kterém lze vytvářet mapy pro hry užívající tento engine. Tato kapitola obsahuje podrobnější rozbor tvorby a editace map v tomto engineu.

6.1 Editace a modifikace

Tato podkapitola se již bude věnovat samotné tvorbě a editaci map pro Doom engine. Jejím cílem je prohloubit pojmy, které byly uvedeny v předchozích kapitolách a poskytnout vhled do struktury map v Doom engineu.

V první podkapitole budou uvedeny zásadní pojmy související s editací v Doom engineu. V podkapitole druhé budou hlouběji představeny parametry objektů probraných v podkapitole první. Ve třetí podkapitole bude pohovořeno o věcech a parametrech, které mají. Ve čtvrté a zároveň poslední podkapitole bude rozveden pojem tag a bude zde pohovořeno o spuštěných akcích v tomto engineu.

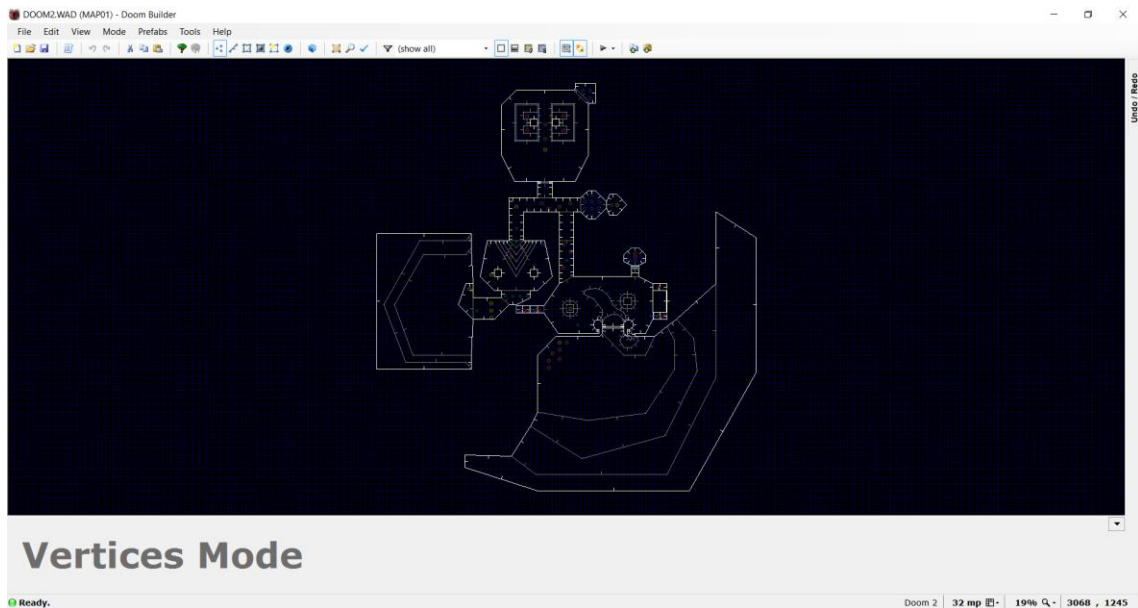
Pro snazší pochopení problematiky zde budou také obrázky pořízené v editoru Doom Builder 2 a v portu pro Doom engine hry ZDoom.

Pokud není specificky uvedeno, veškeré informace obsažené v této kapitole byly zpracovány na základě autorových znalostí a jeho vlastní práce s WAD souborem *doom2.wad*. To se vztahuje i na použité obrázky a tabulky obsahující data obsažené v tomto WAD souboru.

6.1.1 Geometrie map (vertex, linedef, sidedef, sector)

Tvorba map v Doom engineu probíhá v rovině a tomu také odpovídají možnosti a geometrické vlastnosti objektů vertex, linedef a sector. Tvorba tvaru mapy je v podstatě jednoduchá planimetrie sestávající z několika základních geometrických útvarů – z bodů, úseček a mnohoúhelníků. Tyto objekty – vertex, linedef, sidedef a sector – byly vymezeny v **Podkapitole 1.2**.

Nyní budou výše zmíněné geometrické prvky map v Doom engineu představeny za pomoci *MAP01: Entryway z DOOM2.wad*. Budou následovat statistické údaje (tedy počty těchto prvků) a za pomoci obrázků z editoru Doom Builder 2 budou předvedeny příklady jednotlivých prvků.

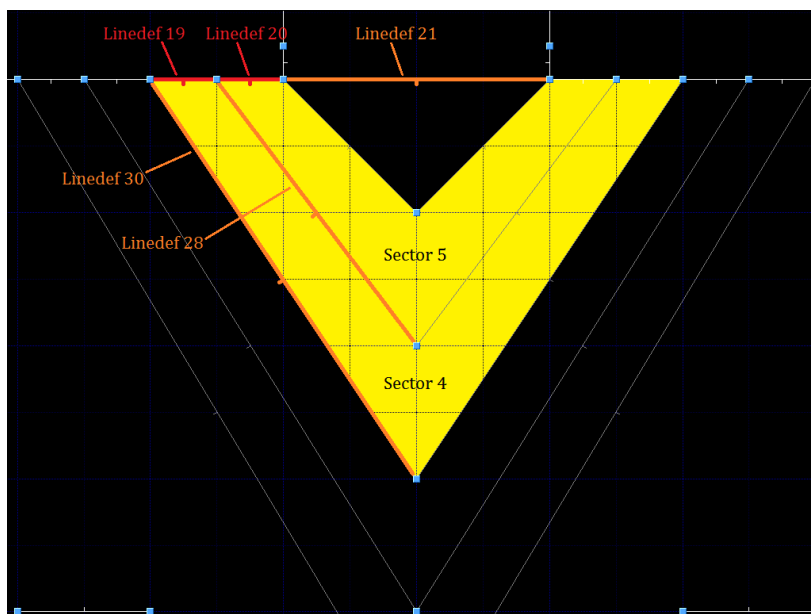


Obrázek 13: MAP01: Entryway

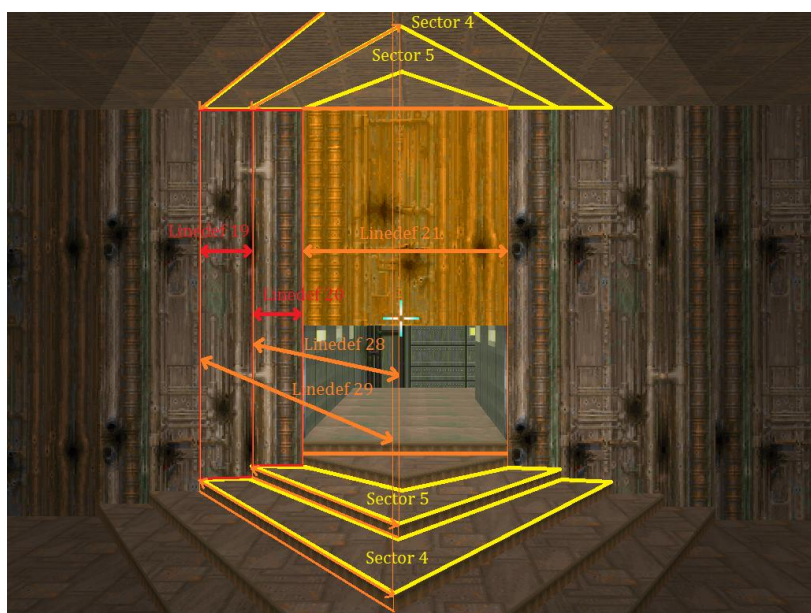


Obrázek 14: MAP01: Entryway – první místnost

Obrázek 13 zachycuje *MAP01: Entryway* otevřenou v editoru Doom Builder 2. **Obrázek 14** zachycuje první místnost téže mapy. Obrázek byl pořízen v portu ZDoom. Tato místnost je první věc, kterou hráč uvidí, když spustí v *Doom II: Hell on Earth* novou hru. Tato místnost navíc efektivně poslouží k demonstraci pojmů v praxi.



Obrázek 15: Entryway – Geometrie ve 2D



Obrázek 16: Entryway – Geometrie ve 3D

Obrázky 15 a 16 byly opět pořizeny v místnosti zachycené na **Obrázku 14**, avšak tentokrát v editoru Doom Builder 2. Pro přehlednost bylo v editoru z místnosti odstraněno celkem pět věcí, které by na **Obrázcích 15 a 16** překážely. Těmito věcmi byly Věci 6 (Former Human), 7 (Former Human), 8 (Ammo clip), 9 (Armor bonus) a 10 (Stimpack). Dále zde byly pomocí programu Malování viditelně zvýrazněny některé geometrické objekty. Jsou jimi dvě červeně zvýrazněné jednostranné úsečky (Linedef 19 a Linedef 20), tři oranžově

zvýrazněné oboustranné úsečky (Linedef 21, Linedef 28 a Linedef 29) a dva žlutě vyplněné/ohraňované ohraničené sektory (Sector 5 a Sector 6).

Na **Obrázku 15**, který je ve 2D módu, jsou vertexy, úsečky a sektory pouze rovinné geometrické útvary, které lze chápat jako půdorys (často nazývaný layout) místnosti na **Obrázku 16**. Na **Obrázku 16**, který je ve 3D módu, je mapě přidán vertikální rozměr.

Sektory, které jsou ve 2D zobrazeny jako určité mnohoúhelníky, se ve 3D stávají hranoly s těmito mnohoúhelníky jakožto základnami. Jestliže se sektor z mnohoúhelníku stane hranolem, úsečky, které sektoru náleží, se stanou jeho stěnami, a vertexy se stanou jeho vertikálními hranami.

6.1.2 Parametry sektorů, úseček a stran úseček

V **Tabulkách 2, 3 a 4** níže budou uvedeny parametry objektů vyznačených na **Obrázcích 15 a 16**. Tyto údaje byly vyčteny buď přímo v editoru Doom Builder 2 v nastavení daného objektu, nebo v programu k prohlížení a editaci WAD souborů XWE v příslušné položce (SECTORS, LINEDEFS, SIDEDEFS apod.).

Tabulka 2: MAP01: Entryway – Sektory 4 a 5

Sector	Floor	Ceiling	Floor T.	Ceiling T.	Light	Type	Tag
Sector 4	32	264	FLOOR0_1	FLOOR4_1	176	0	0
Sector 5	40	264	FLOOR0_1	FLOOR4_1	192	0	0

Tabulka 2 obsahuje parametry Sektorů 4 a 5. Sedm uvedených údajů (sloupec vlevo je zde uveden pouze pro usnadnění orientace) jsou všechna data, která sektor definují. Každý sektor má délku 26 bajtů. Nutno zdůraznit, že v těchto 26 bajtech nejsou obsaženy žádné informace o tvaru sektoru, ten je určen úsečkami a jejich příslušnými stranami.

Sloupce **Floor** a **Ceiling** každý obsahují proměnné typu integer. Tato čísla od -32 768 do 32 767 definují výšky podlahy a stropu daného sektoru. Výška sektoru je pak logicky určena jejich rozdílem.

Sloupce **Floor T.** a **Ceiling T.** obsahují proměnné typu string o délce 8 bajtů. Jsou to názvy textury podlahy a stropu daného sektoru.

Sloupec **Light** opět obsahuje proměnnou typu integer. Ačkoli jsou tomuto parametru vyhrazeny 2 bajty, používají se pouze hodnoty v rozsahu 0 až 255. Tento číselný údaj udává světlost sektoru, přičemž hodnota 0 je nejtmaší a 255 nejsvětlejší. Původní Doom engine navíc při zobrazování světelné úrovně rozlišuje pouze násobky 16, takže zde není možné dosáhnout plynulého světelného přechodu. Sektory 4 a 5 v *MAP01: Entryway* mají světelné hodnoty 176 a 192, takže světelný rozdíl mezi nimi je v podstatě jeden stupeň. Tato stupňovitost světelné úrovně je na **Obrázku 16** dobře pozorovatelná na stropě zachycené místnosti.

Sloupec **Type** obsahuje také proměnnou typu integer. Tento parametr určuje, zda má sektor nějakou speciální vlastnost. Hodnota 0 znamená, že sektor žádnou nemá. Pro tyto speciální vlastnosti jsou pak hodnoty 1 až 17. Hodnoty 1, 2, 3, 8, 12, 13 a 17 jsou různé varianty blikání a pulzování světla. 4, 5, 7 a 16 jsou různé žíravé efekty³. 9 je Secret. 10 je snížení stropu sektoru jako při zavření dveří 30 sekund po spuštění levelu a 14 zvýšení stropu sektoru jako při otevření dveří 5 minut po spuštění levelu. A konečně 11 je žíravý efekt, který po snížení hráčova zdraví pod 11 ukončí level (nejznámější případ využití tohoto typu sektoru je pravděpodobně ukončení *E1M8: Phobos Anomaly* v *doom.wad*).

Sloupec **Tag** také obsahuje proměnnou typu integer. Tento parametr určuje, zda je sektor součástí nějaké spuštěné události, jako je např. dálkové otevření dveří, výtah nebo teleport. Hodnota 0 znamená, že sektor žádný tag nemá, hodnoty vyšší pak označují sektor jako cíl určitých spuštěných událostí, které se spouští obvykle stisknutím spínače nebo přejitím přes určitou úsečku. Toto bude podrobněji rozvedeno v **Podkapitole 6.1.4**.

³ Žíravý efekt se používá obvykle u podlah s texturami kyseliny (napr. NUKAGE1) nebo lávy (např. LAVA1). Podlaha s žíravým efektem na ní stojícího hráče průběžně zraňuje.

Tabulka 3: MAP01: Entryway – Úsečky 19, 20, 28 a 29

Linedef	Start V.	End V.	Flags	Type	Tag	Front	Back
Linedef 19	20	21	1	0	0	21	-1
Linedef 20	21	22	1	0	0	22	-1
Linedef 21	22	23	76	0	0	23	24
Linedef 28	21	28	4	0	0	33	34
Linedef 29	20	29	4	0	0	37	38

Tabulka 3 obsahuje parametry Úseček 19, 20, 21, 28 a 29, které jsou zobrazeny na **Obrázcích 15** a **16**. Těchto sedm hodnot opět definuje celou úsečku (sloupec vlevo je opět uveden pro přehlednost). Všechny tyto hodnoty jsou typu integer, takže mají délku 2 bajty. Celá úsečka pak má délku 14 bajtů. Je třeba zmínit, že samotné úsečky nijak nezmiňují sektory, které ohraničují. Tato náležitost je definována jednotlivými stranami úseček.

Sloupce **Start V.** a **End V.** jsou pořadová čísla vertexů, kterými je úsečka určena. Samotná úsečka tedy není definována souřadnicemi, protože ty jsou obsažené už v parametrech jednotlivých vertexů – a každá úsečka dva takové vertexy spojuje.

Sloupec **Flags** obsahuje číslo určující vlastnosti úsečky zvané flagy. Tyto flagy nemají oficiální jména, ale v Doom Builderu 2 jsou označeny *Impassable*, *Block Monster*, *Double Sided*, *Upper Unpegged*, *Lower Unpegged*, *Secret*, *Block Sound*, *Hidden* a *Shown*. Každému z těchto flagů je vyhrazen 1 bit.

Impassable je neprůchozí úsečka, přes kterou nemůže projít ani hráč, ani nepřítel. *Block Monster* znemožňuje průchod pouze nepřítelům, hráč přes tuto úsečku však může normálně projít. *Double Sided* znamená, že je úsečka oboustranná – náleží jí tedy dvě strany úsečky. *Upper Unpegged* a *Lower Unpegged* jsou flagy, které určují, zda se u stran úsečky budou textury vykreslovat odshora dolů nebo odzdoła nahoru (k *Upper* a *Lower Unpegged* bude více uvedeno u stran úseček). *Secret* znamená, že se úsečka bude ve hře na automapě zobrazovat jako pevná neprůchozí zeď. *Block Sound*, jak název napovídá, znamená, že úsečka blokuje zvuk. Pokud hráč vystřelí a mezi ním a nepřítelem jsou alespoň dvě úsečky s *Block Sound* flagem, nepřítel výstřel neuslyší a neaktivuje se. *Hidden* znamená, že se úsečka ve hře na automapě nebude vůbec zobrazovat a *Shown* naopak znamená, že se na automapě bude zobrazovat vždy, tedy i před tím, než ji hráč odkryje.

Úsečky 19 a 20 mají pouze flag *Impassable*, který má ve Flags hodnotu 1, z čehož vyplývá, že tyto dvě úsečky také mají hodnotu Flags 1. Úsečky 19 a 20 jsou tedy jednostranné úsečky reprezentující neprůchozí zeď. Úsečky 28 a 29 mají pouze flag *Double Sided*, který má hodnotu 4. Jejich hodnota Flags je tedy 4. Úsečky 28 a 29 tedy jsou oboustranné úsečky reprezentující průchozí hranici mezi dvěma sektory. Pakliže by úsečka měla více flagů, výsledná hodnota Flags by byla součtem hodnot těchto flagů – např. úsečka s flagy *Impassable* a *Double Sided* (tedy neprůchozí oboustranná úsečka) by měla hodnotu Flags 5 (1 + 4). Nakonec Úsečka 21 má flagy *Double Sided*, *Upper Unpegged* a *Block Sound*. Její výsledná hodnota Flags je 76 (4 + 8 + 64).

Sloupce **Type** a **Tag** určují, zda bude úsečka spouštěčem nějaké spuštěné události, a pokud ano, jaké (více v **Podkapitole 6.1.4.**).

Sloupce **Front** a **Back** určují, které strany úsečky dané úsečce náleží. Jak je uvedeno v **Tabulce 3**, hodnotou Back může být -1, což znamená, že úsečka žádnou zadní stranu nemá a je tudíž jednostranná.

Tabulka 4: MAP01: Entryway – Strany úseček 21, 22, 33, 34, 37 a 38

Sidedef	X Offset	Y Offset	Above	Below	Main	Sector
Sidedef 21	128	0	-	-	PIPE4	4
Sidedef 22	160	0	-	-	PIPE4	5
Sidedef 23	0	0	PIPE4	STEP2	-	6
Sidedef 24	0	0	-	-	-	52
Sidedef 33	0	0	-	STEP2	-	4
Sidedef 34	0	0	-	-	-	5
Sidedef 37	0	0	-	STEP2	-	3
Sidedef 38	0	0	-	-	-	4

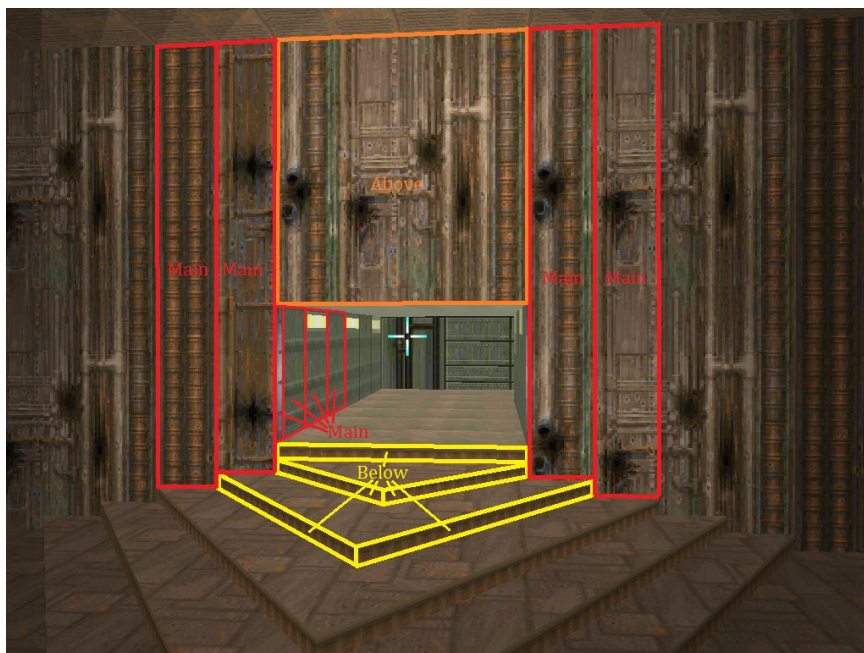
Tabulka 4 obsahuje parametry Stran úseček 21, 22, 23, 24, 33, 34, 37 a 38. Zatímco sektory a úsečky byly definovány sedmi parametry, strany úseček jsou definovány šesti. Vzhledem k tomu, že strany úseček obsahují tři proměnné typu string o délce 8 bajtů, jsou z objektů, kterými se tato kapitola zabývá, největší – dohromady má každá strana úsečky 30 bajtů.

Prvních pět údajů definuje textury, které na určité straně úsečky hráč uvidí. Strany úseček v podstatě zastupují veškerou grafiku, kterou hráč může ve hře na úsečkách vidět. U stran úseček však v této práci bude netradičně nejprve rozveden sloupec poslední, a tím je Sector.

Hodnota **Sector** nese informace o tom, kterému sektoru daná strana úsečky náleží. Ačkoli je na sektor odkazováno pouze ze stran úseček, na strany úseček je odkazováno z úseček samotných. Jsou to tedy úsečky samotné, které určují výsledný tvar sektoru. Pakliže úsečky, které ohraničují sektor, netvoří uzavřený rovinný geometrický útvar, není uzavřený ani sektor samotný a obvykle dochází k chybám. Postupem času však někteří tvůrci map pro Doom engine objevili, že některé z těchto chyb pramenících z neuzavřených sektorů lze využít ve vlastní prospěch.⁴ Díky neuzavřeným sektorům totiž lze vytvořit některé objekty, které Doom engine standardně neumožňuje. Příkladem takového objektu jsou neviditelné podlahy, průhledné dveře nebo tzv. falešné 3D podlahy (fake 3D floors). Tyto postupy při využití bugů v enginu však tvůrce mapy používá záměrně, protože vyžadují přesné uchopení tohoto bugu.

Sloupce **Above**, **Below** a **Main** jsou vyhrazeny pro hodnoty typu string o délce 8 bajtů, které jsou určeny pro názvy textury, které daná strana úsečky má zobrazovat. Main je textura prostřední, Above horní a Below spodní. U jednostranných úseček se zpravidla nastavuje pouze textura prostřední, ostatní textury jsou zde redundantní. Textura horní a spodní se nastavuje u oboustranných úseček, přičemž sektory zmiňované jejich stranami mají rozdílné výšky stropu a podlahy. Co přesně jsou prostřední, horní a spodní textury je zobrazeno na **Obrázku 17**:

⁴ Řadu takových speciálních efektů proslavil finský tvůrce map Iikka Keränen, který jich hojně užíval v *REQUIEM.wad* (*Requiem*) z roku 1997.



Obrázek 17: Entryway – texturey

Na **Obrázku 17** jsou na některých úsečkách barevně označeny příslušné texturey: prostřední (Main) červeně, horní (Above) oranžově a spodní (Below) žlutě.

Jak je patrné, prostřední texturey se používají hlavně na stranách jednostranných úseček, kde udávají texturu dané zdi. Chybějící textura u takové zdi vede k HOM efektu⁵. Prostřední texturey se však dají použít i u úseček oboustranných. V takovém případě se obvykle užívají texturey obsahující průhledné části, jimiž jsou zpravidla různé texturey mříží (např. MIDGRATE), mřížoví oken (např. BRNSMAL1), révy (např. MIDVINE1) apod. Lze však použít i texturey zcela neprůhledné. Je nutné upozornit, že ačkoli takové úsečky mají texturu naznačující její neprůchodnost, úsečka je zcela průchozí. U takových úseček se tedy obvykle nastavuje flag *Impassable*.

Horní a spodní texturey se používají hlavně u stran oboustranných úseček, které jsou hranicí mezi sektory s odlišnými výškami stropu nebo podlahy. V tomto případě je daná textura texturou schodu mezi podlahami nebo stropy těchto dvou sektorů. Pokud takový schod nemá nastavenou texturu, engine na schodu namísto

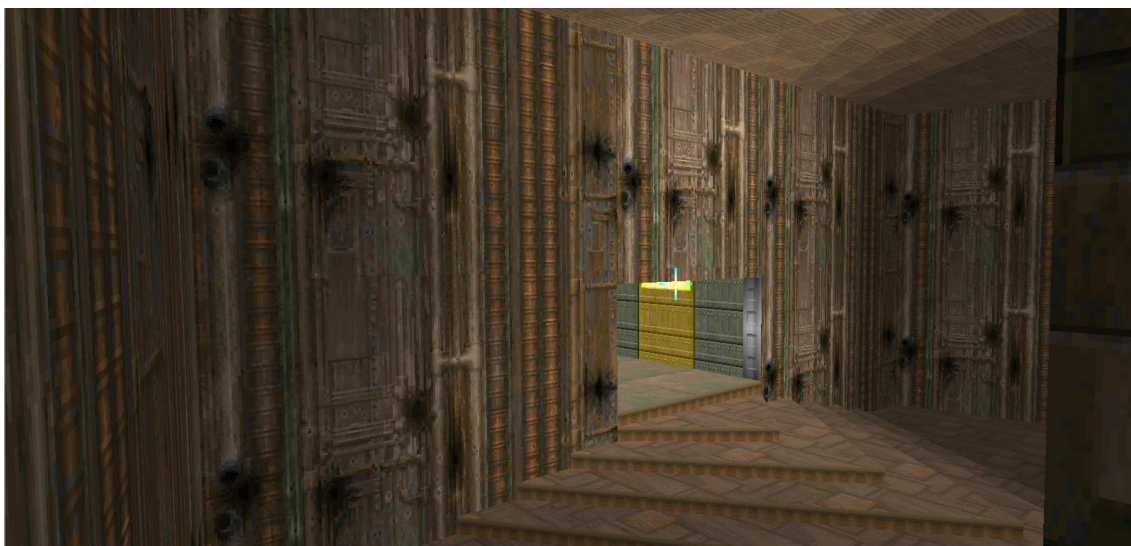
⁵ Hall of Mirrors effect. Protože za jednostrannou úsečkou nic není, engine namísto chybějící texturey zobrazuje prolínající části předchozích snímků.

chybějící textury stěny zobrazuje tzv. krvácející (bleeding) texturu podlahy nebo stropu. To však funguje pouze tehdy, když je textura dané podlahy nebo stropu v zorném úhlu hráče. Pokud tomu tak není, opět dochází k HOM efektu. Pokud úsečka tvoří hranici mezi sektory, které mají odlišnou výšku jak podlahy, tak stropu, daná strana vyžaduje nastavení jak spodní, tak horní textury. Takovou úsečkou v *MAP01: Entryway* je např. Úsečka 21 (viz **Obrázek 16**), jejíž přední stranou je Strana úsečky 23.

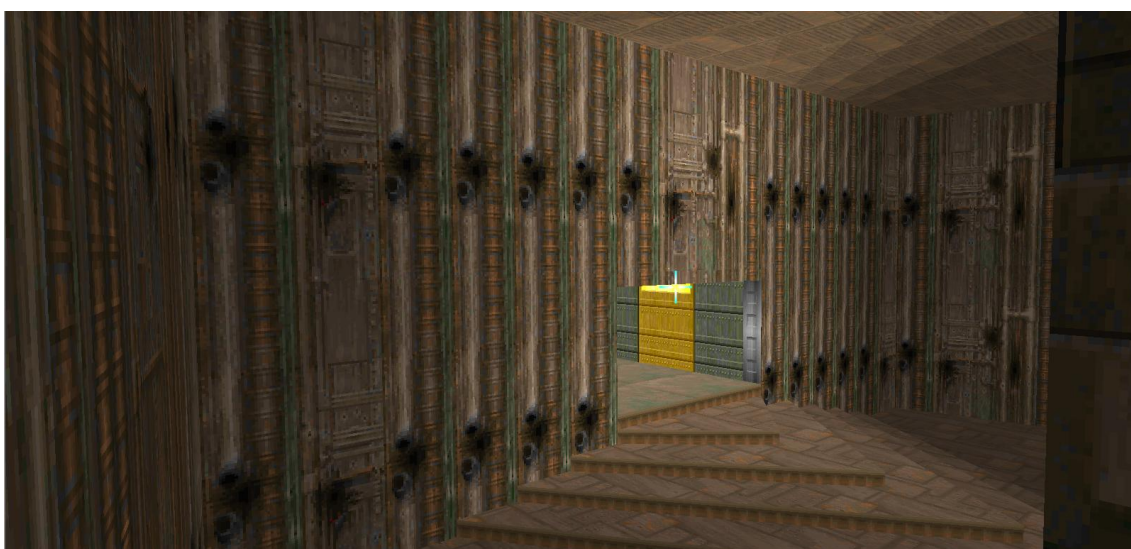


Obrázek 18: HOM efekt na stěně (vlevo) a krvácení textury schodů (vpravo)

X Offset je hodnota odsazení textury po souřadnici X, tedy horizontálně. Tento parametr se užívá, aby na sebe textury na stěnách přirozeně navazovaly. Doom engine totiž na každé straně úsečky standardně načítá textury od začátku. Např. Strany úseček 21 a 22 mají hodnotu X Offset 128 a 160. Protože užívají texturu PIPE4, která má rozměry 256 x 128, je nutné toto odsazení nastavit, jinak by se na každé 32 mapových jednotek dlouhé úsečce, jimiž je tvořena zeď v sektorech tvořících schody, textura načítala opět od začátku a na každé z těchto stěn by tedy bylo pouze prvních 32 pixelů z této 256 pixelů široké textury. Pro snazší pochopení bude tento jev demonstrován pomocí **Obrázků 19** a **20**. Na **Obrázku 19** mají textury odsazení, kterého bylo docíleno pomocí hodnoty X Offset, a tudíž na sebe textury plynule navazují, zatímco na **Obrázku 20** byly tyto hodnoty X Offset nastaveny na 0, takže je textura pro každou úsečku načítána od začátku.



Obrázek 19: Entryway – stěna s X Offsety

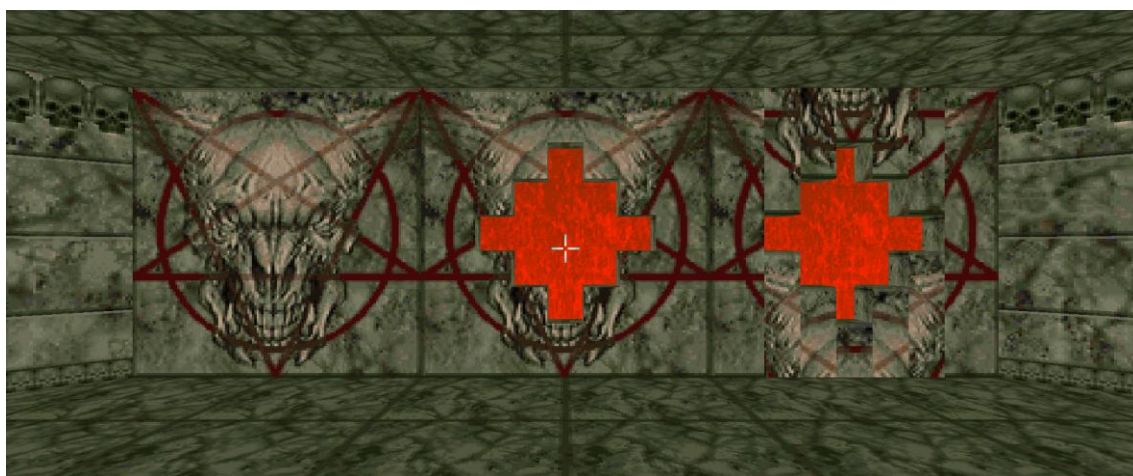


Obrázek 20: Entryway – stěna bez X Offsetů

Y Offset je analogicky hodnota odsazení textury po souřadnici Y, tedy vertikálně. Zatímco u souřadnice X je textura vždy vykreslena zleva doprava, u souřadnice Y směr vykreslení textury, a tedy i směru odsazení pomocí hodnoty Y Offset, závisí na několika faktorech. Prvním faktorem je, zda je úsečka jednostranná, nebo oboustranná. Dále záleží na tom, zda se jedná o texturu prostřední, horní, nebo spodní. Posledním faktorem je použití flagů *Upper Unpegged* a *Lower Unpegged*.

U úsečky jednostranné je jedinou viditelnou texturou textura střední. Tato textura je vykreslena odshora dolů. Pokud má úsečka flag *Lower Unpegged*, textura je vykreslena odzdola nahoru.

U úsečky oboustranné mohou být viditelné všechny tři textury. Pokud úsečka nemá žádný z *Unpegged* flagů, textura horní je vykreslena odzdoła nahoru a textura spodní odshora dolů. Pokud má úsečka flag *Upper Unpegged*, textura horní je vykreslena odshora dolů. Pokud má flag *Lower Unpegged*, textura spodní je vykreslena odzdoła nahoru. Případná textura prostřední je standardně vykreslena odshora dolů, avšak na rozdíl od ostatních textur, které se na dané úsečce ve vertikálním směru, podobně jako tomu bylo u směru horizontálního, opakují, textura prostřední se vykreslí pouze jednou. Pokud má úsečka flag *Lower Unpegged*, vykreslí se odzdoła nahoru, avšak také pouze jednou.



Obrázek 21: Flagy *Upper Unpegged* a *Lower Unpegged*

Na **Obrázku 21** lze dobře vidět, jak flagy *Upper Unpegged* a *Lower Unpegged* ovlivňují vykreslení horních a spodních textur. K této demonstraci byla použita textura *MARBFACE*. Vlevo je celá textura o rozměrech 128 x 128. Uprostřed a vpravo je identický schodovitý otvor, u nějž je textura *MARBFACE* nastavena jako horní i jako spodní textura. Úsečky uprostřed mají flagy *Upper Unpegged* a *Lower Unpegged*, vpravo však ne.

6.1.3 Věci (things)

Ačkoli věci v Doom engine nejsou tématem zdaleka tak rozsáhlým jako geometrie map, je nezbytné jim jednu podkapitolu také věnovat.

Věci jsou nedílnou součástí každé mapy. Věcmi mohou být veškerí nepřátelé, sebratelné předměty, tj. různé zbraně, náboje, lékárny, klíče apod., různé překážky jako jsou např. stromy, sloupy, pochodně apod., či dekorační předměty jako jsou

např. mrtvoly na zemi, visící mrtvoly, svíčky apod. Je důležité, že mezi věci patří i startovní pozice hráče, bez níž level ani nejde spustit. Mezi věci patří také projektily vystřelené nepřáteli nebo hráčem, ty však nelze do mapy přímo umístit. Některé z výše uvedených věcí jsou zobrazeny na **Obrázku 22**:



Obrázek 22: Entryway – věci

Obrázek 22 zachycuje pohled na již dobře známou úvodní místnost *MAP01: Entryway*. Na obrázku lze v popředí vidět čtyři věci reprezentující startovní pozice hráčů 1–4. Na schodech lze vidět dvě monstra Former Human. V chodbě lze vidět startovní pozici pro deathmatch. Thing 0–3 jsou pořadová čísla daných věcí v mapě, která značí, v jakém pořadí byly tyto věci do mapy umísťovány. Na funkčnost dané věci toto pořadové číslo nemá vliv⁶, jedná se pouze o parametr, pomocí něž engine identifikuje danou věc.

Ve WAD souborech jsou věci uloženy v položce THINGS. Stejně jako byly v předchozí podkapitole uvedeny výňatky z příslušných položek náležitých různým geometrickým objektům tvořícím mapu, bude i v této podkapitole ilustrační tabulka s parametry několika věcí z *MAP01: Entryway*. Sedm z jedenácti vybraných věcí i bylo označeno na **Obrázku 22**.

⁶ Výjimku tvoří vícenásobné startovní pozice Player 1–4 start. V takovém případě hráč začíná na startovní pozici s nejvyšším pořadovým číslem, přičemž na ostatních startovních pozicích je také vytvořen avatar hráče, který nelze ovládat. Tento avatar lze zranit, což však zraní hráče samotného. Tento avatar, pokud je např. střelbou donucen k pohybu, může sbírat předměty nebo spouštět spuštěné akce. Takové startovní pozice se v komunitě ze zjevných důvodů označují pojmem "Voodoo doll".

Tabulka 5: MAP01 Entryway – Věci 0, 1, 2, 3, 6, 7, 8, 9, 10, 57 a 60

Thing	X Position	Y Position	Angle	Type	Flags
Thing 0	-96	784	90	1	7
Thing 1	-32	784	90	2	7
Thing 2	-160	784	90	3	7
Thing 3	32	784	90	4	7
Thing 6	-112	1136	90	3004	7
Thing 7	-16	1216	90	3004	7
Thing 8	-304	1248	270	2007	7
Thing 9	176	1248	270	2015	7
Thing 10	96	1056	270	2011	7
Thing 57	-288	976	270	2006	23
Thing 60	-64	1392	90	11	7

Tabulka 5 obsahuje parametry Věcí 0, 1, 2, 3, 6, 7, 8, 9, 10, 57 a 60. Těchto pět hodnot definuje každou z věcí (sloupec vlevo je opět uveden pro přehlednost). Všechny z těchto hodnot jsou typu integer, takže mají délku 2 bajty. Každá věc má tedy délku 10 bajtů. Dále budou vysvětleny parametry, které jsou uvedeny v **Tabulce 5**.

Sloupce **X Position** a **Y Position** udávají souřadnice dané věci na mapě. Na stejných souřadnicích může být současně i více věcí.

Angle udává směr, kterým je daná věc při spuštění levelu natočena. Výchozí hodnota 0 znamená, že je věc natočena na východ, 90 na sever, 180 na západ a 270 na jih. Dalšími používanými hodnotami jsou 45, 135, 225 a 315. Angle má vliv pouze na některé věci, konkrétně na startovní pozice, nepřátele a na Teleport Destination⁷. U ostatních věcí tento parametr nehraje žádnou roli.

Type udává typ věci. Z **Obrázku 22** a **Tabulky 5** lze vyvodit, že typ 1 je Player 1 start, typ 2 Player 2 start, typ 3 Player 3 start a typ 4 Player 4 start. Stejně tak lze odvodit, že typ 3004 je Former Human a 11 Player Deathmatch start. Kromě nich jsou v Tabulce 5 zmíněny typy 2006, 2007, 2015 a 2011. To jsou BFG9000, Ammo clip, Armor bonus a Stimpack.

⁷ Česky "cíl teleportu". Na této pozici se octne hráč nebo nepřítel, když projde teleportem s příslušným tagem.

Flags je číslo, které, podobně jako u úseček, je součtem flagů dané věci. Na rozdíl od úseček mají věci pouze pět flagů, které jsou v Doom Builderu 2 nazvány *Easy*, *Medium*, *Hard*, *Deaf* a *Multiplayer*. Každému z těchto flagů je opět vyhrazen 1 bit.

Flagy *Easy*, *Medium* a *Hard* určují, zda je věc v levelu přítomná při dané obtížnosti. *Easy* je pro obtížnosti 1 a 2, *Medium* pro 3 a *Hard* pro 4 a 5. Flag *Deaf* (česky "hluchý") se nastavuje u nepřátel, kteří se nemají aktivovat, když slyší hráčův výstřel, ale pouze tehdy, když hráče uvidí. U ostatních věcí je tento flag redundantní. Flag *Multiplayer* se používá, pokud má být věc přítomna pouze při deathmatchi nebo cooperative módu.

Stejně jako u úseček, i u věcí je výsledná hodnota parametru *Flags* určena součtem příslušných bitů. Deset z jedenácti věcí zmíněných v Tabulce 5 mají hodnotu *Flags* 7 (1 + 2 + 4). To znamená, že jsou přítomny při obtížnostech *Easy*, *Medium* i *Hard*. Věc 57 má hodnotu *Flags* 23 (1 + 2 + 4 + 16). Je tedy přítomna na všech obtížnostech, ale pouze při deathmatchi nebo cooperative módu.

6.1.4 Spuštěné události (tag, action)

Tagy jsou parametry úseček a sektorů (parametr *Tag*) a akce jsou parametrem úseček (parametr *Type*). Vzhledem k relativní komplexnosti spuštěných událostí byla této oblasti věnována celá podkapitola.

Jako spuštěné události jsou v této práci označovány veškeré akce, které nějak dynamicky mění vlastnosti objektů v mapách. Jednodušeji řečeno, tato kapitola se bude zabývat všemožnými dveřmi, výtahy, teleporty, výsuvnými plošinami a dalšími dynamickými objekty, které se v mapách v Doom engine objevují.

Pojmy tag a akce jsou zásadní. Tag (česky "visačka", "štítek" nebo jednoduše "značka") je parametrem, který se nastavuje jak úsečkám, tak sektorům. Používá se, aby engine dokázal identifikovat, ke kterému sektoru nebo sektorům se daná akce vztahuje. Akce se nastavuje úsečkám a jedná se o typ akce, který se má při určité události aplikovat na sektor nebo sektory s příslušným tagem.

Protože se akce nastavují pouze úsečkám, je nasnadě, že jediným způsobem, jak akci spustit, je právě interakce s úsečkou. Nelze tedy vytvořit akci, která by se spustila např. po vstupu hráče do daného sektoru.

Dále existuje několik akcí, které k fungování nepotřebují tag a stejně tak nepotřebují sektor se stejným tagem.

Vzhledem k tomu, že akce jsou ve WAD souborech ukládány pouze jako čísla, budou zde používána názvosloví z editoru Doom Builder 2.

V **Tabulce 6** je uvedeno několik významnějších akcí:

Tabulka 6: Seznam vybraných akcí

Číslo akce	Třída	Spuštění	Efekt
0	None	-	-
1	Door	DR	Door Open Wait Close (also monsters)
11	Exit	S1	Exit Level
26	Door	DR	Door (Blue) Open Wait Close
31	Door	D1	Door Open Stay
36	Floor	W1	Floor Lower to 8 above Highest Floor
46	Door	GR	Door Open Stay
48	Scroll	-	Scroll Texture Left
52	Exit	W1	Exit Level
62	Lift	SR	Lift Lower Wait Raise
73	Crusher	WR	Crusher Start with Slow Damage
74	Crusher	WR	Crusher Stop
83	Floor	WR	Floor Lower to Highest Floor
91	Floor	WR	Floor Raise to Lowest Ceiling
97	Teleport	WR	Teleport
103	Door	S1	Door Open Stay
109	Door	W1	Door Open Stay (fast)
123	Lift	SR	Lift Lower Wait Raise (fast)
126	Teleport	WR	Teleport (monsters only)

Jak lze vyčíst z **Tabulky 6**, v editoru Doom Builder 2 jsou akce pro přehlednost rozříděny do několika tříd: *Ceiling*, *Crusher*, *Door*, *Exit*, *Floor*, *Lift*, *Light*, *Scroll*, *Stairs* a *Teleport*. Mezi akcemi v **Tabulce 6** je většina z těchto tříd zastoupena.

Každá akce má určitý způsob spuštění. Dvojice znaků ve sloupci Spuštění určují právě tento způsob. **D** znamená "Door", **S** "Switch", **W** "Walk-over" a **G** "Gun". Každá akce je pak použitelná buď jen jednou, nebo opakovaně, podle čehož je označena buď číslem **1** (one-time), nebo písmenem **R** (repetitive).

Akce se spuštěním **D** se ve hře spouští stejně jako akce **S** pomocí klávesy "Use". Tyto akce jsou však výjimečné tím, že nepoužívají tagy. Engine identifikuje sektor, který se má po stisku dané úsečky otevřít, podle zadní strany úsečky. Pokud je akce **D** nastavena na jednostranné úsečce, po jejím použití se nic nestane.

Akce se spuštěním **S** již tag vyžadují. Po stisku dané úsečky je u všech sektorů s příslušným tagem provedena daná akce. Výjimku tvoří exit, který, ačkoli má spuštění **S**, tag nevyžaduje.

Akce se spuštěním **W** se spustí, pokud hráč překročí danou úsečku. Takové akce tedy nemá smysl nastavovat úsečkám jednostranným nebo neprůchozím. Po překročení úsečky se provede daná akce u všech sektorů s příslušným tagem. Výjimku opět tvoří exit, který, ačkoli má spuštění **W**, tag nevyžaduje.

Akce se spuštěním **G** se spustí, pokud hráč strelí do dané úsečky. Taková úsečka může být jednostranná i oboustranná. Po výstřelu do ní se provede daná akce u všech sektorů s příslušným tagem.

7 Quake engine

Zatímco předchozí kapitola představila tvorbu map v Doom enginu, tato kapitola se bude věnovat Quake enginu. Tato kapitola bude mít podobnou strukturu jako kapitola předchozí a slouží hlavně k porovnání obou enginů.

7.1 Editace a modifikace

Tato podkapitola se již bude věnovat samotné tvorbě a editaci map pro Quake engine. Jejím cílem je hlavně poskytnout základní vhled do struktury map v Quake enginu.

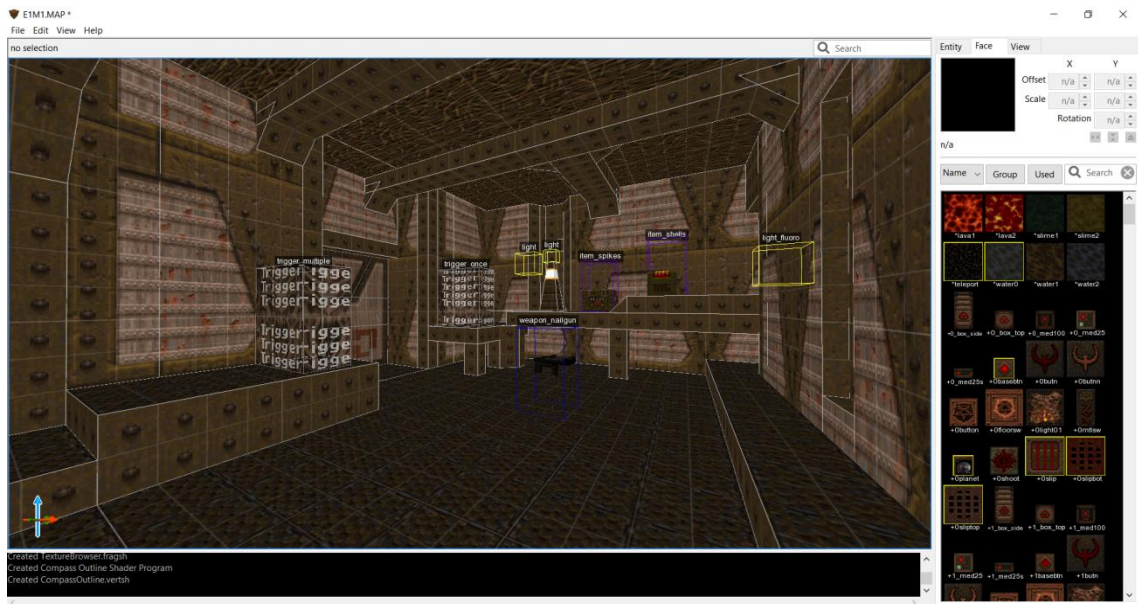
V první podkapitole opět budou podrobněji rozvedeny zásadní pojmy související s editací v Quake enginu. V podkapitole druhé bude pohovořeno o entitách a jejich parametrech. Ve třetí podkapitole bude pohovořeno o entitách spuštěných událostí. Ve čtvrté a zároveň poslední podkapitole bude stručně rozvedeno chování světla v Quake enginu.

Pro snazší pochopení problematiky zde budou také obrázky pořízené v editoru TrenchBroom a v portu pro hru *Quake DirectQ*.

Pokud není specificky uvedeno, veškeré informace obsažené v této kapitole byly zpracovány na základě autorových znalostí a jeho vlastní práce s MAP souborem *E1M1.map*. To se vztahuje i na použité obrázky a ukázky kódu z tohoto MAP souboru.

7.1.1 Geometrie map (brush, face)

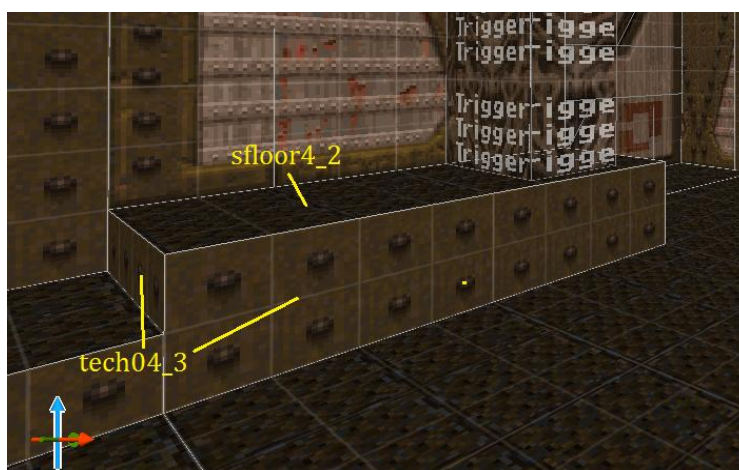
Jak už bylo několikrát zmíněno, Quake engine je plně 3D a tomu také odpovídá geometrie map. Zatímco v případě Doom enginu pro jeho tzv. 2.5D prostředí veškerá geometrie byla dvojrozměrná (objekty měly pouze souřadnice X a Y) a třetí rozměr mapa dostávala až díky parametrům jednotlivých objektů, v Quake enginu objekty mají skutečně tři rozměry (objekty mají kromě souřadnic X a Y i souřadnici Z). Tyto objekty – brush a face – byly vymezeny v **Podkapitole 1.3**.



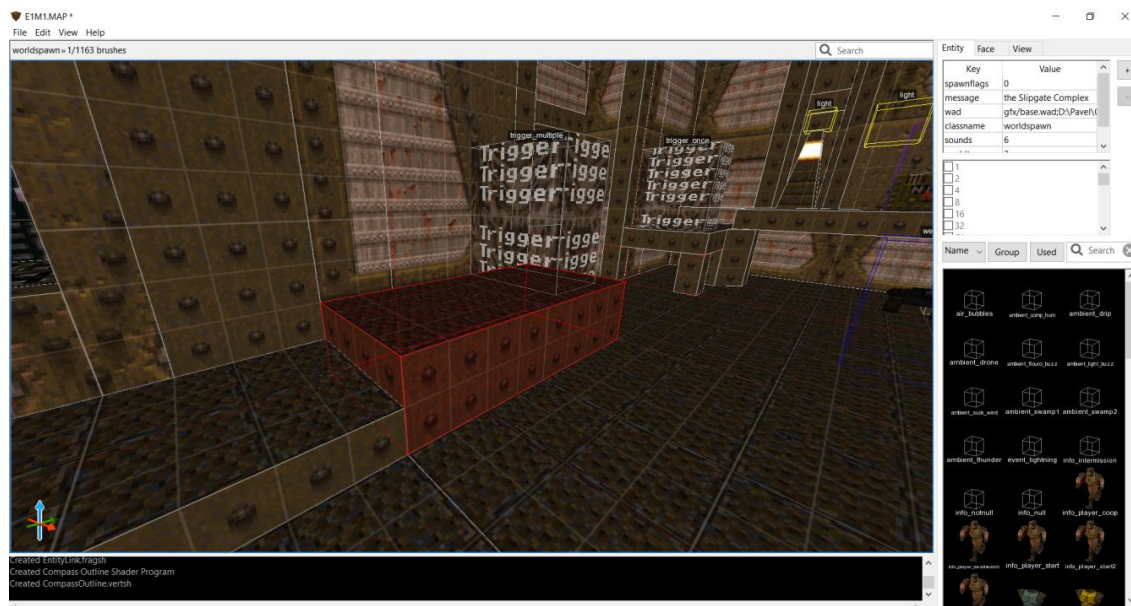
Obrázek 23: E1M1: The Slipgate Complex – první místnost

Obrázek 23 zachycuje první místnosti mapy *E1M1: The Slipgate Complex* otevřené v editoru TrenchBroom. Vzhledem ke způsobu, kterým TrenchBroom mapu zobrazuje, zde není nutné nijak náročně označovat různé geometrické objekty. Bílé čáry totiž zřetelně označují jejich hrany. Na obrázku je velký počet různých brushů a pochopitelně ještě větší počet stěn.

Pro ilustraci, např. kvádrovitý útvar v levé spodní části obrázku je jeden brush a každá z jeho stěn (v geometrickém slova smyslu) je také stěnou v kontextu Quake enginu. Lze si všimnout, že různé stěny náležící jednomu brushu mohou mít různé textury. V případě tohoto brushe můžeme vidět dvě stěny s texturou `tech04_3` a jednu s texturou `sfloor4_2` (viz **Obrázek 24**).



Obrázek 24: Textury `tech04_3` a `sfloor4_2`



Obrázek 25: The Slipgate Complex – brush

Na **Obrázku 25** je tento brush označen, díky čemuž lze vidět i jeho stěny, které na **Obrázcích 23** a **24** zůstávaly skryty.

Definice tohoto brushe v souboru *E1M1.map* je následující:

```
{
( 320 512 32 ) ( 320 384 32 ) ( 320 384 16 ) sfloor4_2 -0 -0 -0 1 1
( 384 512 32 ) ( 320 512 32 ) ( 320 512 16 ) tech04_3 -0 -0 -0 1 1
( 384 384 32 ) ( 384 512 32 ) ( 384 512 16 ) tech04_3 -0 -0 -0 1 1
( 320 384 32 ) ( 384 384 32 ) ( 384 384 16 ) tech04_3 -0 -0 -0 1 1
( 320 384 32 ) ( 320 512 32 ) ( 384 512 32 ) sfloor4_2 -0 -0 -0 1 1
( 384 512 -0 ) ( 320 512 -0 ) ( 320 384 -0 ) sfloor4_2 -0 -0 -0 1 1
}
```

Složené závorky ohraničují brush, resp. ohraničují stěny, jimiž je daný brush tvořen. Každý řádek pak je definicí jedné stěny. Tato definice obsahuje trojici kulatých závorek obsahujících pokaždé tři číselné hodnoty. To jsou souřadnice příslušného bodu určujícího stěnu. Jak již bylo zmíněno, stěny jsou de facto roviny, a rovinu ve trojrozměrném prostoru lze určit třemi body o souřadnicích X, Y a Z. Po číselných údajích následuje jméno textury dané stěny. Nakonec následuje pět parametrů, které se mají na texturu aplikovat: *x_off* (odsazení textury po ose X), *y_off* (odsazení textury po ose Y), *rot_angle* (úhel otočení textury), *x_scale* (roztážení textury v ose X) a *y_scale* (roztážení textury v ose Y). [15]

7.1.2 Entity (entities)

Entity jsou nedílnou součástí map v Quake engine. Na první pohled entity zastávají roli, kterou mají věci v Doom engine, avšak entity v Quake engine jsou pojmem mnohem širším. Je pravda, že veškerí nepřátelé, sebratelné předměty apod. jsou entitami, avšak mezi entity patří i tzv. **brush entities** a hlavně entita **worldspawn**⁸.

Worldspawn je nejdůležitější entitou mapy v Quake engine a každá taková mapa tuto entitu musí obsahovat. Worldspawn je obvykle v MAP souboru hned na prvním místě. [15] Entita worldspawn totiž obsahuje definice všech statických brushů v mapě. Tudíž i brush, jehož definice byla uvedena v předchozí podkapitole, je součástí entity worldspawn obsažené v souboru *E1M1.map*.

Entita worldspawn může vypadat např. takto (uvedený příklad pochází z *E1M1.map*):

```
{
"spawnflags" "0"
"message" "the Slipgate Complex"
"wad" "gfx/base.wad"
"classname" "worldspawn"
"sounds" "6"
"worldtype" "2"
...
}
```

Na místě tří teček by pak byly veškeré statické brushe obsažené v mapě. Například *E1M1.map* takových statických brushů obsahuje 1163, přičemž drtivá většina z nich má podobnou nebo dokonce větší délku než v předchozí podkapitole uvedený příklad. Z toho lze vyvodit, že položka entity worldspawn je velmi rozsáhlá a zpravidla zabírá většinu MAP souboru.

Mezi entity se dále řadí brush entities, což jsou brushe, které mají přiřazené nějaké speciální a spuštěné akce. Těmito entitami se bude hlouběji zabývat následující podkapitola.

⁸ Nejvýstižnějším českým ekvivalentem by zřejmě bylo "zrození světa".

Mezi entity patří objekty jako startovní pozice, nepřítelé, sebratelné předměty apod. Následuje několik příkladů takových entit:

```
{
"spawnflags" "0"
"angle" "90"
"origin" "480 -352 88"
"classname" "info_player_start"
}
```

```
{
"spawnflags" "768"
"classname" "monster_army"
"origin" "616 72 40"
"angle" "180"
}
```

```
{
"spawnflags" "0"
"origin" "-360 2912 -80"
"classname" "weapon_supershotgun"
}
```

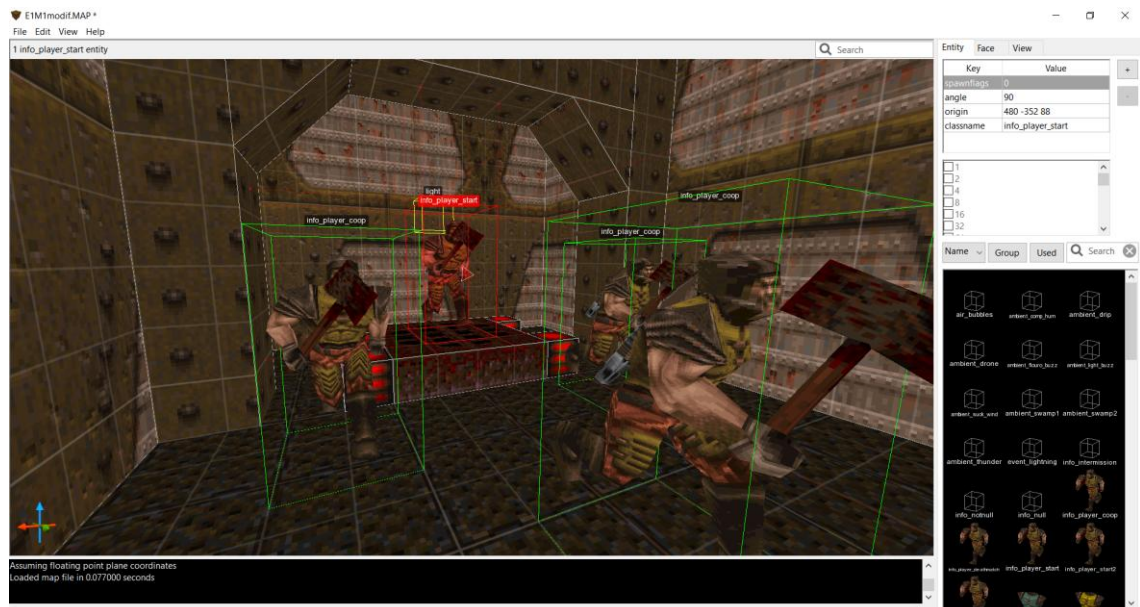
Části kódu, které jsou uvedeny výše, opět pochází z *E1M1.map*. Jedná se o položky, které v této mapě definují startovní pozici hráče (`info_player_start`), jeden exemplář nepřítele typu Grunt (`monster_army`) a dvouhlavňovou brokovnici (`weapon_supershotgun`).

Classname je parametr, který definuje, o entitu jakého typu se jedná, **origin** určuje souřadnice dané entity a **angle** je úhel určující směr natočení dané entity. **Spawnflags** je, podobně jako flagy u věcí v Doom engine, číslo získané sečtením flagů dané entity. U uvedených příkladů se vyskytují hodnoty 0 a 768. Hodnota 0 znamená, že entita nemá přiřazené žádné flagy. Hodnota 768 u zmíněného exempláře nepřítele typu Grunt značí, že entita má přiřazené flagy *Not on Easy* a *Not on Normal* (256 + 512). U různých entit mohou dané flagy mít jiný efekt.

Nyní budou ještě zmíněny některé další parametry, kterými disponovala entita `worldspawn` na předešlé stránce. Parametr **message** u entity `worldspawn` určuje

jméno levelu po jeho spuštění. Parametr **wad**⁹ určuje cestu ke zdrojovému souboru s texturami používanými v dané mapě. Parametr **sounds** v případě entity worldspawn určuje, jaká hudební stopa se má při hraní levelu přehrávat. Nakonec parametr **worldtype** určuje prostředí v levelu, čímž je ovlivněn vzhled klíčů. Těmito prostředími může být hodnota 0 pro prostředí *Medieval*, 1 pro *Runic (metal)* a 2 pro *Present (base)*. [15]

Entity mohou mít i jiné parametry, které zde nebyly uvedeny, a stejně tak stejné parametry u různých entit mohou mít jinou funkci.¹⁰



Obrázek 26: The Slipgate Complex – startovní pozice

Obrázek 26 zachycuje startovní pozice hráčů pro singleplayer a pro cooperative mód v mapě *E1M1: The Slipgate Complex*. Startovní pozice hráče pro singleplayer je na obrázku označena a v pravé horní části obrázku lze vidět tabulku s parametry této entity. Jedná se o entitu uvedenou mezi příklady výše.

Mezi entity také patří zdroje světla (viz **Podkapitola 7.1.4**).

⁹ Ačkoli Quake engine pracuje s jinými soubory než Doom engine, soubory typu WAD se stále užívají jako zdrojový soubor obsahující textury, který je potřeba při editaci a při následné kompilaci souboru typu MAP do souboru typu BSP. WAD soubory používané Quake engineem však nejsou kompatibilní s Doom engineem a naopak.

¹⁰ Takovým příkladem může být např. parametr message. V případě entity worldspawn tento parametr určuje jméno levelu, zatímco u většiny jiných entit při určité události zobrazí mírně nad středem obrazovky text obsažený v tomto parametru.

7.1.3 Entity spuštěných událostí

Zatímco v Doom engine byly spuštěné události vždy předem definovány a byly vždy určeny pouze tagem spouštěcí úsečky a cíleného sektoru a pak samotnou akcí, která se měla provést, v Quake engine je tvorba takových událostí komplexnější, ale tvůrce mapy zato získává relativní volnost.

V předešlé podkapitole již zazněl pojem **brush entity**. Jedná se o entitu tvořenou definicí samotné entity a pak definicí jednoho nebo více brushů, které dané entitě náleží a se kterými bude akce při spuštění pracovat. Tyto entity nejsou součástí entity worldspawn.

Následuje příklad takové brush entity, jíž jsou dveře z první místnosti mapy *E1M1: The Slipgate Complex*, nebo přesněji jejich levé křídlo, přičemž pravé křídlo by vypadalo obdobně, s tím rozdílem, že by pochopitelně obsahovalo jiný úhel a jiné brushe:

```
{
"spawnflags" "0"
"classname" "func_door"
"angle" "270"
"sounds" "2"
"speed" "400"
{
( 256 464 -0 ) ( 256 592 -0 ) ( 208 592 -0 ) door02_1 0 0 0 1 1
( 256 512 44 ) ( 256 512 -84 ) ( 208 512 -84 ) edoor01_1 0 0 0 1 1
( 256 600 48 ) ( 256 472 48 ) ( 208 472 48 ) door02_1 0 0 0 1 1
( 256 600 48 ) ( 256 600 -80 ) ( 256 472 -80 ) edoor01_1 0 0 0 1 1
( 208 472 -80 ) ( 208 600 -80 ) ( 208 600 48 ) edoor01_1 0 0 0 1 1
( 208 572 48 ) ( 208 544 -0 ) ( 224 572 48 ) door02_1 0 0 0 1 1
}
{
( 256 464 80 ) ( 256 592 80 ) ( 208 592 80 ) door02_1 0 0 0 1 1
( 256 512 124 ) ( 256 512 -4 ) ( 208 512 -4 ) edoor01_1 0 0 0 1 1
( 256 600 128 ) ( 256 472 128 ) ( 208 472 128 ) edoor01_1 0 0 0 1 1
( 256 600 128 ) ( 256 600 -0 ) ( 256 472 -0 ) edoor01_1 0 0 0 1 1
( 208 472 -0 ) ( 208 600 -0 ) ( 208 600 128 ) edoor01_1 0 0 0 1 1
( 208 616 128 ) ( 208 588 80 ) ( 224 616 128 ) door02_1 0 0 0 1 1
}
{
( 256 538 48 ) ( 208 538 48 ) ( 208 512 48 ) edoor01_1 0 0 0 1 1
( 256 512 72 ) ( 256 512 -56 ) ( 208 512 -56 ) edoor01_1 0 0 0 1 1
( 256 600 80 ) ( 256 472 80 ) ( 208 472 80 ) edoor01_1 0 0 0 1 1
}
```

```

( 256 600 76 ) ( 256 600 -52 ) ( 256 472 -52 ) edoor01_1 0 0 0 1 1
( 208 472 -52 ) ( 208 600 -52 ) ( 208 600 76 ) edoor01_1 0 0 0 1 1
( 208 538 48 ) ( 256 538 48 ) ( 256 556 76 ) door02_1 0 0 0 1 1
}
}

```

Jak lze vidět v jejím zápisu, entita dveří se jmenuje "func_door". Tato třída určuje, o jaký objekt, resp. o jakou akci se jedná. Tyto konkrétní dveře mají tři parametry, jimiž jsou **angle**, **sounds** a **speed**. **Angle** je úhel, kterým se dveře otevírají. 0 je jako obvykle otevírání na východ, 90, na sever, 180 na západ a 270 na jih. Jako hodnota pro angle se dále používá -1 pro dveře, které se otevírají kolmo vzhůru a -2 pro dveře, které se otevírají kolmo dolů. **Sounds** je zvuk, který se má při spuštění akce přehrát. **Speed** je rychlost, kterou se má entita při spuštění akce posunout – v případě dveří se jedná o rychlost jejich otevření.

Dalšími významnými třídami entit spuštěných událostí jsou např. "func_plat" pro výtah a "func_button" pro spínač.

Kromě entit třídy "func", které vždy nějak manipulují se svými brushi, se v Quake engine ještě vyskytují entity třídy "trigger". Brushe takových entit při spuštění levelu nejsou viditelné a hráč a nepřátelé jimi mohou normálně procházet, avšak při průchodu jimi se spustí nějaká akce. Fungují tedy podobně jako úsečky s akcí se spouštěčem **W** v Doom engine.

Pokud entita trigger má sloužit jako dálkový spouštěč, je třeba ji propojit s jinou entitou, která je obvykle typu func, aby engine poznal, jaký objekt má po aktivaci spouštěče svou akci provést. Za tímto účelem se používají parametry **target** pro spouštěč a **targetname** pro cílenou entitu. Interakcí s triggerem s určitým parametrem target se spustí akce všech entit se stejným parametrem targetname.

Jako spouštěče mohou v Quake engine fungovat i entity různých nepřátel nebo sebratelných předmětů. Takovým entitám se taktéž přiřazuje příslušný parametr target. V případě nepřítele se akce spustí po jeho smrti a v případě sebratelného předmětu po jeho sebrání hráčem.

Quake engine také umožňuje vytvářet sekvence spínačů nebo jiných spouštěčů. Typickým příkladem takové sekvence je sekvence tří spínačů v levelu *E1M1: The Slipgate Complex*, kde hráč musí stisknout tři spínače, aby otevřel dveře, přičemž

po stisknutí každého hra hráči oznámí, kolik spínačů zbývá (např. *"Only 2 more to go..."*) a po dokončení sekvence hra oznámí *"Sequence completed!"*.

K takovým sekvencím se užívá entita "trigger_counter". Spínače mají jako parametr target jméno příslušné entity trigger_counter, která pak má jako parametr target jméno entity, která akci provede (např. dveře). Kromě parametrů target a pochopitelně targetname pak entita trigger_counter obvykle má parametr count, který určuje počet spínačů, které hráč musí stisknout, aby došlo k dokončení sekvence. Trigger_counter se běžně používá i k vytváření sekvencí vyžadujících smrt několika nepřátel.

Na závěr, na rozdíl od Doom engine, kde akce byla vždy parametrem spouštěcí úsečky, v Quake engine je akce určena cílovou entitou. Spouštěč vyžaduje pouze parametr target, aby cílovou entitu identifikoval. Podmínka pro spuštění je pak podmíněna třídou spouštěče – brush spustí akci při jeho projití hráčem, entita nepřítele spustí akci při své smrti apod.

Jeden spouštěč může současně spustit několik různých událostí, a stejně tak provedení události může samo spustit události další.

7.1.4 Světla

Zatímco v Doom engine se nastavovalo osvětlení jako parametr sektoru, Quake engine disponuje entitami fungujícími jako zdroje světla. Díky tomu v Quake engine světlo působí mnohem realističtěji: přechody mezi světlem a stínem jsou jemnější, s rostoucí vzdáleností od svého zdroje se světlo pozvolna vytrácí a překážky stojící světlu v cestě vrhají stín. Světelné rozdíly se v Quake engine, opět na rozdíl od Doom engine, projevují i na kolmých stěnách. V Doom engine totiž kvůli způsobu nastavení osvětlení takové světelné přechody možné nebyly – hodnota osvětlení na kolmých stěnách byla identická odshora až dolů.

Následuje obrázek zachycující jednoduchou místnost vytvořenou za účelem demonstrace možností a chování světla v Quake engine:



Obrázek 27: Světlo v Quake enginu

Obrázek 27 zachycuje čtvercovou místnost obsahující krychlovou krabici s plamenem uvnitř. Tento plamen je jediným zdrojem světla v celé mapě a stíny na podlaze, stropu a stěnách jsou vrženy sloupky, které světlu brání v průchodu.

Následuje definice entity plamene z této mapy:

```
{  
"spawnflags" "0"  
"classname" "light_flame_large_yellow"  
"origin" "192 192 76"  
"angle" "-0"  
"light" "512"  
}
```

Třída tohoto plamene se jmenuje "light_flame_large_yellow". Kromě běžných parametrů je zde i parametr **light**. Tento parametr u entit světelných zdrojů určuje současně intenzitu a dosah světla vycházejícího z entity.

Ke světlu v Quake engine je ještě záhodno zmínit, že stíny vrhají pouze brushe obsažené v entitě worldspawn, entity jako dveře nebo plošiny výtahu stín nevrhají. To platí i pro entity jako je hráč, přičemž nepřátelé či různé další objekty stín také nevrhají.

Veškeré osvětlení a tím pádem i stíny jsou v Quake enginu vypočítány předem. K tomuto výpočtu dochází během kompilace mapy z formátu MAP na BSP. Jedním z takových programů na výpočty světla je *light.exe*. Pokud mapa neobsahuje žádné světelné zdroje, po kompilaci je všude v mapě maximální osvětlení.

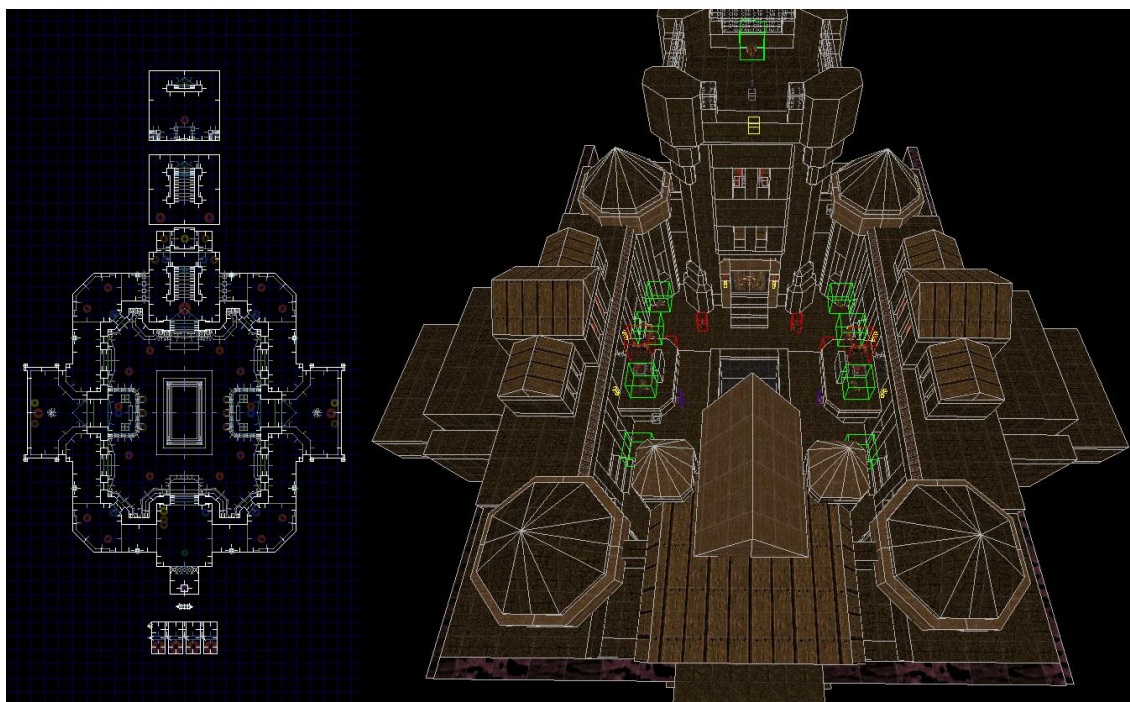
8 Vlastní vytvořené mapy v Doom a Quake enginech

Tato kapitola je porovnáním dvou map, které vytvořil autor práce specificky pro její účely. První mapa (soubor *bc_doom.wad*) je pro Doom engine. Druhá mapa (soubor *bc_quake.map* pro nezkompilovanou mapu a *bc_quake.bsp* pro její zkompilovanou hratelnou verzi) je vylepšenou konverzí *bc_doom.wad* pro Quake engine. Vzhledem ke značné odlišnosti editace map v obou enginech byla však i tato mapa vytvářena zcela od začátku. Z pochopitelných důvodů v této mapě také byli použiti nepřátelé a předměty ze hry *Quake*.

Mapa pro Quake engine obsahuje řadu prvků, které Doom engine nepodporuje. Tyto prvky a jejich rozdílnost budou hlavním obsahem kapitoly.

Kapitola obsahuje obrázky, které autor práce pořídil v editorech Doom Builder 2 a TrenchBroom a v portech ZDoom a DirectQ. Účelem těchto obrázků je posloužit k porovnání obou map a jejich příslušných enginů.

8.1 Pohled na mapy v editorech



Obrázek 28: Pohled na mapy v editorech

Obrázek 28 zachycuje obě mapy otevřené v příslušných editorech. Mapa vlevo je mapa *bc_doom.wad* otevřená v editoru Doom Builder 2 a mapa vpravo je mapa *bc_quake.map* otevřená v editoru TrenchBroom.

Na první pohled lze zaznamenat, že mapa pro Doom engine je v podstatě pouhým půdorysem, jehož třetí rozměr je získán až nastavením jednotlivých sektorů, zatímco mapa pro Quake engine je plně trojrozměrná i v editoru.

Z obrázku také lze vyvodit, že zatímco mapa pro Quake engine na mnohých místech obsahuje více pater nad sebou, šikmé plošiny a celkově vyšší úroveň detailu, zvláště pak na střechách, mapa pro Doom engine nic takového neobsahuje. Z důvodu limitace Doom engine na jedno patro nebylo možné vytvořit věž s více patry, a proto autor musel tuto část mapy vytvořit pomocí více nepropojených místností, mezi nimiž se hráč teleportuje. Tyto místnosti jsou dvě čtvercové místnosti v severní části mapy zachycené na **Obrázku 28**.

8.2 Srovnání map

Tato podkapitola je srovnáním obou map z pohledu hráče ve hře. Tohoto porovnání je docíleno za pomoci osmi obrázků (**Obrázky 29–36**), přičemž na obrázcích vlevo je vždy screenshot z mapy pro Doom engine pořízený v portu ZDoom a vpravo screenshot z mapy pro Quake engine pořízený v portu DirectQ. Po každém obrázku vždy následuje odstavec, který stručně porovná rozdíly, které lze na příslušném obrázku vidět.



Obrázek 29: Srovnání 1 - Dveře

Obrázek 29 zachycuje dveře v první místnosti. V Doom engine se dveře otevírají zásadně kolmo vzhůru, zatímco v Quake engine se mohou otevírat pod různými úhly. V tomto případě jsou dveře dvojdílné a otevírají se do stran.



Obrázek 30: Srovnání 2 - Mříž

Obrázek 30 zachycuje mříž naproti dveřím v první místnosti. V Doom engine byla mříž vytvořena pouze jako několik vertikálních sloupků, zatímco v Quake engine je možné vytvořit i horizontální mříže. Tyto mříže se stejně jako dveře na předchozím obrázku otevírají do stran.



Obrázek 31: Srovnání 3 - Knihovna

Obrázek 31 zachycuje knihovnu v západní části mapy. Na obrázku lze vidět lucernu visící ze stropu. V Doom engine byla vytvořena pomocí úseček s texturami vytvořenými autorem specificky pro takovéto lucerny, zatímco v Quake engine je lucerna složena z několika brushů. Lze si všimnout, že v Quake engine je plamen v lucerně zdrojem světla a sloupky a spodek lucerny vrhají stín. Další odlišností je výtah vpravo od hráče. V Doom engine se jedná o sektor, který se při aktivaci sníží. Hráč takový výtah může vnímat pouze jako kvádrovitý blok. V Quake engine se jedná o plošinu, přičemž jak nad ní, tak pod ní, je volný prostor.



Obrázek 34: Srovnání 6 - Věž - první patro

Obrázek 34 zachycuje pohled na schodiště v prvním patře věže. V Doom engineu pod schodištěm není žádný prostor, schodiště je ohraničeno zdmi a schodiště ve skutečnosti nevede do druhého patra, ale vede to teleportu, který hráče přemístí do místnosti, která má druhé patro představovat. V Quake engineu pod schodištěm je volný prostor a secret s teleportem na střechu věže. Schodiště není obklopeno pevnou zdí, ale zábradlím, a schodiště skutečně vede do druhého patra věže.



Obrázek 35: Srovnání 7 - Věž - druhé patro

Obrázek 35 zachycuje pohled ze severozápadního rohu druhého patra věže. V Doom engineu je teleport, který vede do nižšího patra věže, a za zdí se nachází další schodiště, které je opět zakončeno teleportem do další místnosti, která představuje třetí patro. V Quake engineu lze vidět cestu do nižšího patra. Namísto zdi ohraničující schody je zde zábradlí, a za ním lze vidět schodiště do dalšího patra.



Obrázek 36: Srovnání 8 - Věž - třetí patro

Obrázek 36 zachycuje pohled do místnosti ve třetím patře věže. V Doom engine je podlaha pevná a v místnosti není nic zvláštního k vidění. V Quake engine však lze v pravé dolní části obrázku vidět zábradlí a část průchodu do nižšího patra. Pochodně byly vymodelovány z několika brushů a plameny v nich jsou zdroji světla. Oknem je vidět ven z věže.

Závěr

V této bakalářské práci byl nejprve v teoretické části práce uveden stručný vhled do problematiky herních 3D enginů a jejich historického vývoje. Dále bylo stručně pojednáno o společnosti id Software a jejím působením na tomto poli, zvláště pak o historii herních enginů z řady id Tech. Poté byly za pomoci obrázků a přehledu vlastností podrobněji představeny stěžejní herní tituly této společnosti, a to *Wolfenstein 3D*, *Doom* a *Quake*.

V praktické části práce byly za přispění autorových vlastních znalostí a zkušeností s editací a modifikací v těchto enginech tyto enginy rozebrány a probrány detailněji, přičemž poslední kapitola praktické části práce porovnála přiložené mapy v *Doom* enginu a v *Quake* enginu, které byly vytvořeny autorem této práce specificky za tímto účelem.

Práce měla za cíl čtenáře seznámit hlavně se dvěma výše uvedenými herními enginey, které byly obrovským přínosem ve vývoji herního průmyslu a 3D grafiky celkově. Oba enginy a jejich zástupci *Doom* a *Quake* se navzdory svému stáří i v dnešní době těší relativní popularitě a oblíbenosti a stále kolem nich existuje aktivní komunita hráčů a tvůrců map a modifikací.

Dle autorova názoru je však důležité, aby minimálně o existenci a historické důležitosti těchto dvou herních titulů věděla i široká společnost, a aby se na ně a na počítačové hry všeobecně pohlíželo i jako na kulturní a technologický přínos, jímž bezpochyby jsou.

Autorovým přáním je, aby tato bakalářská práce dovedla eventuální čtenáře k takovému uvažování a pohledu na tuto problematiku, a případně ve čtenáři i vyvolala hlubší zájem o ni. Dokonce zde autor vidí i možnost určitého didaktického přínosu, kterého by bylo možné dosáhnout seznámením žáků či studentů s možnostmi editace map, což je nejen způsob, jak prohloubit jejich technické znalosti, ale i způsob, jak procvičit jejich představivost a fantazii a celkově vzbudit zájem o informatiku.

Seznam použité literatury

- [1] KUSHNER, David. *Masters of Doom: How Two Guys Created an Empire and Transformed Pop Culture*. New York: Random House, 2003. ISBN 0375505245.
- [2] BSP (file format). In: *Wikipedia, the Free Encyclopedia* [online]. San Francisco (CA): Wikimedia Foundation, 2001- [cit. 21. 05. 2016]. Dostupné z: <[https://en.wikipedia.org/wiki/BSP_\(file_format\)](https://en.wikipedia.org/wiki/BSP_(file_format))>.
- [3] Game engine. In: *Wikipedia, the Free Encyclopedia* [online]. San Francisco (CA): Wikimedia Foundation, 2001- [cit. 21. 05. 2016]. Dostupné z: <https://en.wikipedia.org/wiki/Game_engine>.
- [4] List of Unreal Engine Games. In: *Wikipedia, the Free Encyclopedia* [online]. San Francisco (CA): Wikimedia Foundation, 2001- [cit. 20. 05. 2016]. Dostupné z: <https://en.wikipedia.org/wiki/List_of_Unreal_Engine_games>.
- [5] Products. *Havok* [online]. ©2015 [cit. 20. 05. 2016]. Dostupné z: <<http://www.havok.com/products>>.
- [6] LEWIS, Michael a Joffrey JACOBSON. Game Engines in Scientific Research. *Communications of the ACM*. 2002, 45(1), 27-31. ISSN 0001-0782.
- [7] List of first-person shooter engines. In: *Wikipedia, the Free Encyclopedia* [online]. San Francisco (CA): Wikimedia Foundation, 2001- [cit. 20. 05. 2016]. Dostupné z: <https://en.wikipedia.org/wiki/List_of_first-person_shooter_engines>.
- [8] id Software. In: *Wikipedia, the Free Encyclopedia* [online]. San Francisco (CA): Wikimedia Foundation, 2001- [cit. 20. 05. 2016]. Dostupné z: <https://en.wikipedia.org/wiki/Id_Software>.
- [9] CARMACK, John a Doomworld. *Doomworld -- Interviews* [online]. ©1999 [cit. 29. 06. 2016] Dostupné z: <<http://www.doomworld.com/interviews/jnt7.shtml>>.
- [10] Quake engine. In: *Wikipedia, the Free Encyclopedia* [online]. San Francisco (CA): Wikimedia Foundation, 2001- [cit. 21. 05. 2016]. Dostupné z: <https://en.wikipedia.org/wiki/Quake_engine>.
- [11] id Tech 3. In: *Wikipedia, the Free Encyclopedia* [online]. San Francisco (CA): Wikimedia Foundation, 2001- [cit. 20. 05. 2016]. Dostupné z: <https://en.wikipedia.org/wiki/Id_Tech_3>.

- [12] id Tech 4. In: *Wikipedia, the Free Encyclopedia* [online]. San Francisco (CA): Wikimedia Foundation, 2001- [cit. 20. 05. 2016]. Dostupné z: <https://en.wikipedia.org/wiki/Id_Tech_4>.
- [13] id Tech 5. In: *Wikipedia, the Free Encyclopedia* [online]. San Francisco (CA): Wikimedia Foundation, 2001- [cit. 20. 05. 2016]. Dostupné z: <https://en.wikipedia.org/wiki/Id_Tech_5>.
- [14] WAD. In: *The Doom Wiki at DoomWiki.org* [online]. [cit. 22. 05. 2016]. Dostupné z: <<http://doomwiki.org/wiki/WAD>>.
- [15] *Quake MAP Specs* [online]. [cit. 22. 05. 2016]. Dostupné z: <<http://www.gamers.org/dEngine/quake/QDP/qmapspec.html>>.
- [16] NAYLOR, Bruce F. *A Tutorial on Binary Space Partitioning Trees* [online]. [cit. 29. 06. 2016]. Dostupné z: <http://dip.sun.ac.za/~henri/bsp_tutorial.pdf>.
- [17] *Doom Builder map editor* [online]. ©2008 [cit. 22. 05. 2016]. Dostupné z: <<http://www.doombuilder.com>>.
- [18] *TrenchBroom* [online]. [cit. 22. 05. 2016]. Dostupné z: <<http://kristianduske.com/trenchbroom>>.

Seznam příloh

Příloha 1: Přílohy\bc_doom.wad (Soubor WAD)

Příloha 2: Přílohy\bc_quake.bsp (Soubor BSP)

Příloha 3: Přílohy\bc_quake.map (Soubor MAP)