

Analys av spelmotorer för visualisering

Pierre Nerbring

Examensarbete

LUTMDN-TMAT-5144-SE

EAT2010

Institutionen för Designvetenskaper

Lunds Tekniska Högskola



TYRÉNS

LUNDS
UNIVERSITET

Innehåll

1	ABSTRACT	4
1.1	ENGLISH	4
1.2	SVENSKA	4
2	FÖRORD	5
3	INLEDNING	6
3.1	BAKGRUND	6
3.2	PRESENTATION AV FÖRETAGET	6
3.3	PROBLEMSTÄLLNING	7
3.4	MÅLSÄTTNING	7
3.5	FOKUS OCH AVGRÄNSNINGAR	7
4	ARBETSGÅNG	9
4.1	FALLSTUDIE – VÄSTRA LÄNKEN	9
4.1.1	<i>Cryengine 2</i>	12
4.1.2	<i>Unity3D</i>	17
4.2	FALLSTUDIE – HUS FRÅN AUTODESK REVIT ARCHITECTURE	20
4.2.1	<i>Cryengine2</i>	21
4.2.2	<i>Unity3D</i>	22
5	UTVÄRDERING AV SPELMOTORERNA	24
5.1	LICENSFORM OCH KOSTNADER	24
5.1.1	<i>Cryengine 2</i>	24
5.1.2	<i>Unity3D</i>	24
5.2	HUR LÄTT ÄR DET ATT FÖRÄNDRA GUI	24
5.2.1	<i>Cryengine 2</i>	24
5.2.2	<i>Unity3D</i>	25
5.3	IMPORTERING	27
5.3.1	<i>Cryengine 2</i>	27
5.3.2	<i>Unity3D</i>	28
5.4	VAD FÖLJER MED VID IMPORT	30
5.4.1	<i>Cryengine 2</i>	30
5.4.2	<i>Unity3D</i>	30
5.5	EXPORT AV ANIMERING/BILD	32
5.5.1	<i>Cryengine 2</i>	32
5.5.2	<i>Unity3D</i>	32
5.6	PRESTANDA/HÅRDVARUKRAV	33
5.6.1	<i>Cryengine 2</i>	33
5.6.2	<i>Unity3D</i>	33
5.7	ANDRA ANVÄNDBARA FUNKTIONER	34
5.7.1	<i>Cryengine 2</i>	34
5.7.2	<i>Unity3D</i>	36
6	MANUAL FÖR IMPORT	37

6.1	CRYENGINE	37
6.1.1	<i>Installation av 3ds Max, motorn och nödvändiga plugins.....</i>	37
6.1.2	<i>Inställningar i 3ds Max efter du installerat motorn</i>	37
6.1.3	<i>Exportera material och texturer.....</i>	38
6.1.4	<i>Exportera modeller från 3ds Max.....</i>	38
6.1.5	<i>Importera modeller till Cryengine</i>	39
6.1.6	<i>Lägga på en högupplöst textur på terrängen.....</i>	39
6.2	UNITY.....	40
7	DISKUSSION	41
7.1	CRYENGINE	41
7.2	UNITY.....	41
8	SLUTSATS	43
9	KÄLL- OCH LITTERATURFÖRTECKNING	44

1 Abstract

1.1 English

One of the many assignments a infrastructure consulting company has is to visualize the finale result. This aims to provide the customer with a sense of how the structures will look like in the real world. The simple way of doing so is to render a picture or a video of the result. To provide the user with a more interactive and real experience is to present the model through a game engine. This enables many new functions such as being able to navigate in the environment with a virtual character, drive a car on a road or to interact with doors and elevators in a building.

This paper will include an analysis of the process of importing a 3D model from 3ds Max to two different game engines. The engines that will be analyzed are Unity 3d and Cryengine2. This paper will also include information about pricing and licencing, requirements and the features supported by the different engines.

1.2 Svenska

En av de många uppdrag ett konsultföretag inom infrastruktur har är att visualisera det slutgiltiga resultatet. Syftet med det är att ge kunden en känsla av hur byggnaderna kommer se ut i verkligheten tillsammans med den befintliga miljön. Det enkla sättet är att rendera bilder och filmer av resultatet. För att ge användaren en mer interaktiv och verklig upplevelse kan man presentera modellen genom en spelmotor. Detta möjliggör många nya funktioner som att t ex kunna navigera i miljön med en virtuell karaktär, köra en bil på en planerad väg eller att interagera med dörrar och hissar i en byggnad.

Denna rapport innehåller en analys av processen för att importera en 3D-modell från 3ds Max till två olika spelmotorer. De motorer som är analyserade är Unity 3d och Cryengine2. Denna rapport innehåller även information om prissättning och licenser, krav och vilka funktioner som stöds av motorerna.

2 Förord

Denna rapport är ett resultat av examensarbetet som jag har utfört mellan juni och november 2010. Genom min handledare Joakim Eriksson som jag träffat genom kurserna Användbarhetsutvärdering och Virtual reality i teori och praktik på LTH fick jag kontakt med Jörgen Nätterlund på företaget Tyréns. Han var nyfiken på hur användbart en spelmotor kan vara som ett verktyg för visualisering. Eftersom jag är intresserad av både interaktionsdesign och visualisering valde jag att ta uppdraget att analysera detta område. Eftersom Jörgen inte arbetade på kontoret i min närhet blev två andra personer som arbetar på kontoret i Malmö delaktiga i mitt arbete; Pär Hagberg och Anders Dahl. De har hjälpt mig med bland annat tekniken i 3D-labbet och med mjukvaruproblem. Jag har även fått hjälp med spelmotorn Cryengine av Stefan Isberg som jobbar med ett projekt på LTH där de importerar CAD-modeller till Cryengine.

Jag vill tacka:

- alla på Tyréns i Malmö för all hjälp och för att fått mig att känna mig välkommen på företaget.
- Jörgen Nätterlund för du tagit dig tid med projektet.
- Joakim Eriksson för visat stort intresse och för all teknisk hjälp.

3 Inledning

3.1 Bakgrund

En av de många uppdrag ett konsultföretag inom infrastruktur har är att visualisera det slutgiltiga resultatet. Syftet med det är att ge kunden en känsla av hur byggnaderna kommer se ut i verkligheten tillsammans med den befintliga miljön. Det enkla sättet är att rendera bilder och filmer av resultatet. För att ge användaren en mer interaktiv och verklig upplevelse kan man presentera modellen genom en spelmotor. Detta möjliggör många nya funktioner som att t ex kunna navigera i miljön med en virtuell karaktär, köra en bil på en planerad väg eller att interagera med dörrar och hissar i en byggnad.

Vissa spelmotorer kan dessutom simulera olika väderförhållanden och ha en mer dynamisk kontroll av ljussättning som t ex en förinställd bana som solen ska följa. Man kan då göra skugganalyser för olika tidpunkter och eftersom allting renderas i realtid kan man göra ändringar efter kunders förfrågan under en presentation.

3.2 Presentation av företaget

Tyréns AB är ett av Sveriges ledande konsultföretag inom samhällsbyggnad. De erbjuder kundanpassade tjänster inom områdena stadsbyggnad och infrastruktur samt byggnads- och fastighetssektorn. Deras verksamhet är rikstäckande, överallt med tillgång till Tyréns samlade kompetens.

De har sedan starten 1942 utvecklats till en koncern med omkring 1000 medarbetare på ett 20-tal kontor runtom i landet. Tyréns är drivande inom forskning och utveckling och har ett nära samarbete med universitet och högskolor. Deras ambition är att skapa bättre samhällen för människor. (Tyréns Website - Om Tyréns, 2010)

3.3 Problemställning

Det finns ett flertal problemställningar att behandla när man vill bygga upp en virtuell värld med objekt både från verkligheten och från datorgjorda modeller. Det största problemet är att export och import ska fungera mellan programmen. Representationer av byggnader, träd, vatten och vägar ska placeras ut på korrekt position. Terrängen ska byggas upp som den är i verkligheten med hjälp av en heightmap, alltså en svart/vit bild vars pixlar representerar en viss höjd på landskapet. Den ska även ha rätta texturer som skapas av flygfoton.

Förutom nämnda problem finns det några andra funktioner som ska testas:

- Kan man modifiera användargränssnittet och till vilken grad?
- Vad följer med vid import, med tanke på texturkoordinater, materialegenskaper, färger, lager, linjer och animeringar?
- Vad finns det för möjlighet till att använda plattformen till export av bilder och tidssynkade animeringar?
- Kan man simulera olika väderförhållanden och tidpunkter?
- Finns det stöd för multiplayer-funktion?
- Finns det för stöd för styrenheter som t ex 3D-mus, handkontroll och ratt?

3.4 Målsättning

Målet med projektet är att producera en utförlig analys samt att bygga upp några demonstrationsexempel. Analysen ska resultera i ett svar på frågan om vilken spelmotor som lämpar sig bäst att använda för detta ändamål.

3.5 Fokus och avgränsningar

Mitt uppdrag i detta examensarbete var att analysera Unity, Cryengine och UDK (Unreal Development Kit). Efter jag använt spelmotorerna och fått en känsla av hur de fungerar ville jag fokusera på Unity och Cryengine. Anledning till detta var att

Cryengine och UDK är snarlika på många punkter såsom grafisk prestanda och portabilitet och jag kunde inte hitta några fördelar med UDK som Cryengine inte hade. Dessutom hade jag svårt att komma igång med UDK på grund av det dåliga gränssnittet och de importproblem jag möttes av. Därför inriktade jag mig på Cryengine och Unity och det är anledningen till att UDK inte är med i denna analysen.

4 Arbetsgång

Arbetsgången har delats in i fyra olika faser. Den första fasen var att undersöka och dokumentera problemställningar, krav och vilken information och material som behövs för att bygga upp miljön. Detta gjordes mestadels med hjälp av mina handledare på skolan och på Tyréns eftersom jag inte hade så mycket erfarenhet av visualisering inom samhällsbyggnad.

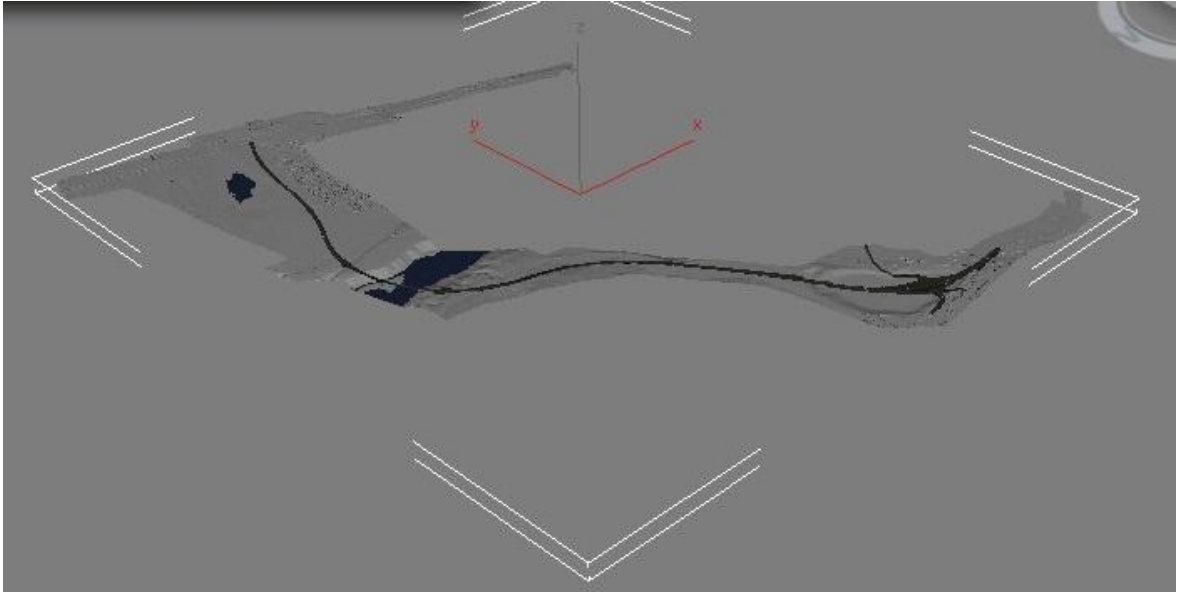
Den andra fasen var att köpa och installera spelmotorerna och samla information om hur importen skulle gå till. Alla nödvändiga mjukvaror installerades på en dator på Tyréns och för att hantera alla filer som behövdes för till projektet, skapade jag en gemensam mapp där alla inblandade kunde lägga upp och ta del materialet.

Nästa fas var att experimentera med import till spelmotorerna. Det var här det mesta av resurserna lades. Uppgiften här var att ta reda på vad man kan importera, vilka program man kan importera från, vad som följer med och vad som inte följer med vid import.

Den fjärde fasen var att skapa en miljö för att kunna demonstrera spelmotorerna och visa hur slutresultaten kommer se ut. Under hela arbetsgången har jag antecknat mina resultat för att få en mer genomgående och kronologisk projektrapport.

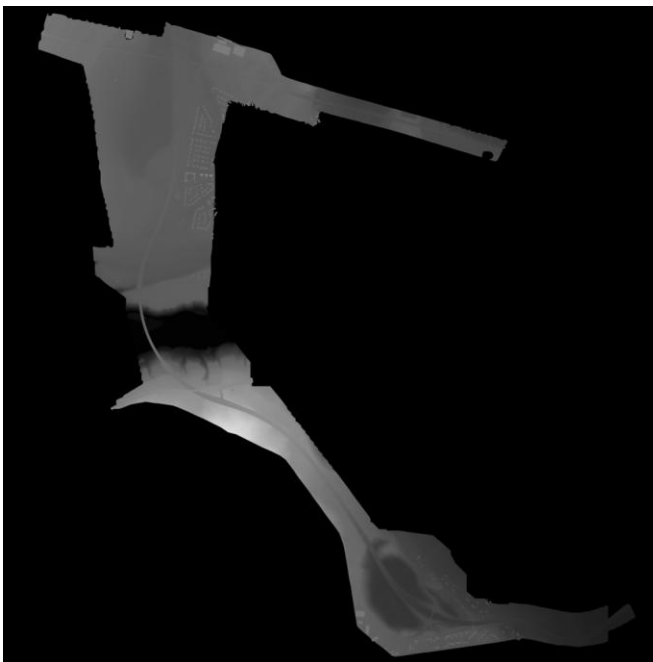
4.1 Fallstudie – Västra länken

Detta var en studie för att undersöka hur motorerna hanterar stora terränger och texturer. Jag hade en 3ds Max-modell över ett område där en ny väg och en bro ska byggas (se figur1).



Figur 1: 3D-modell i 3ds Max.

Jag hade även en heightmap som är genererad från modellen (se figur 2) och ett ortofoto över området (se figur 3).

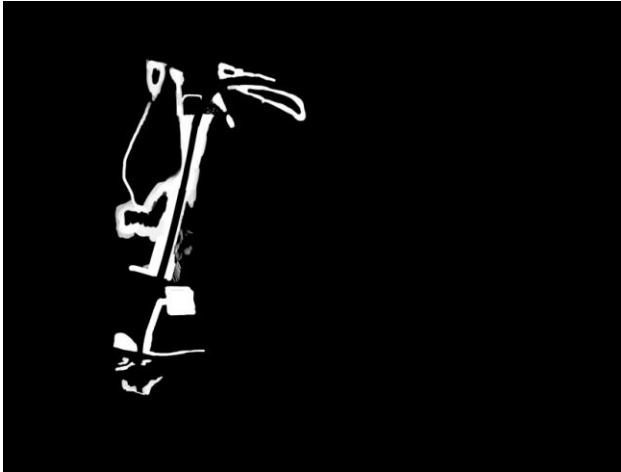


Figur 2: Heightmap



Figur 3: Ortofoto

Dessutom skulle byggnader och växtlighet placeras in i miljön. För att få växtligheten på rätt plats använde jag mig av en mask för träd (se figur 4) och en mask för undervegetation (se figur 5).



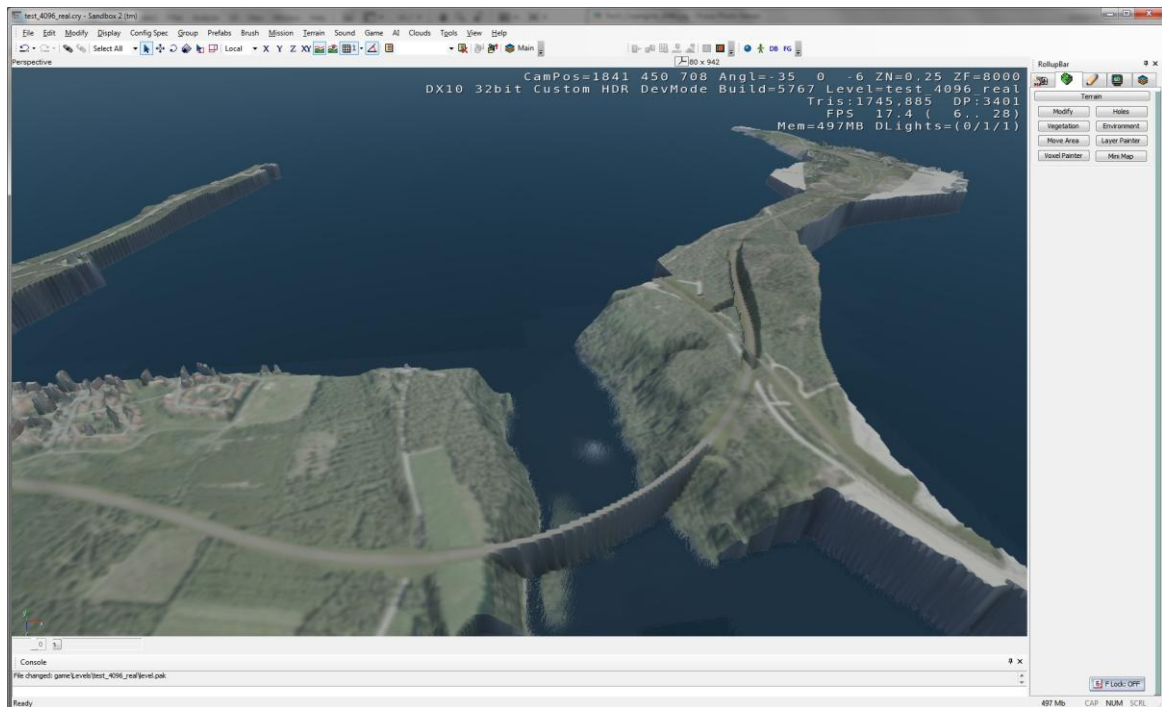
Figur 4: Trädmask



Figur 5: Undervegetation

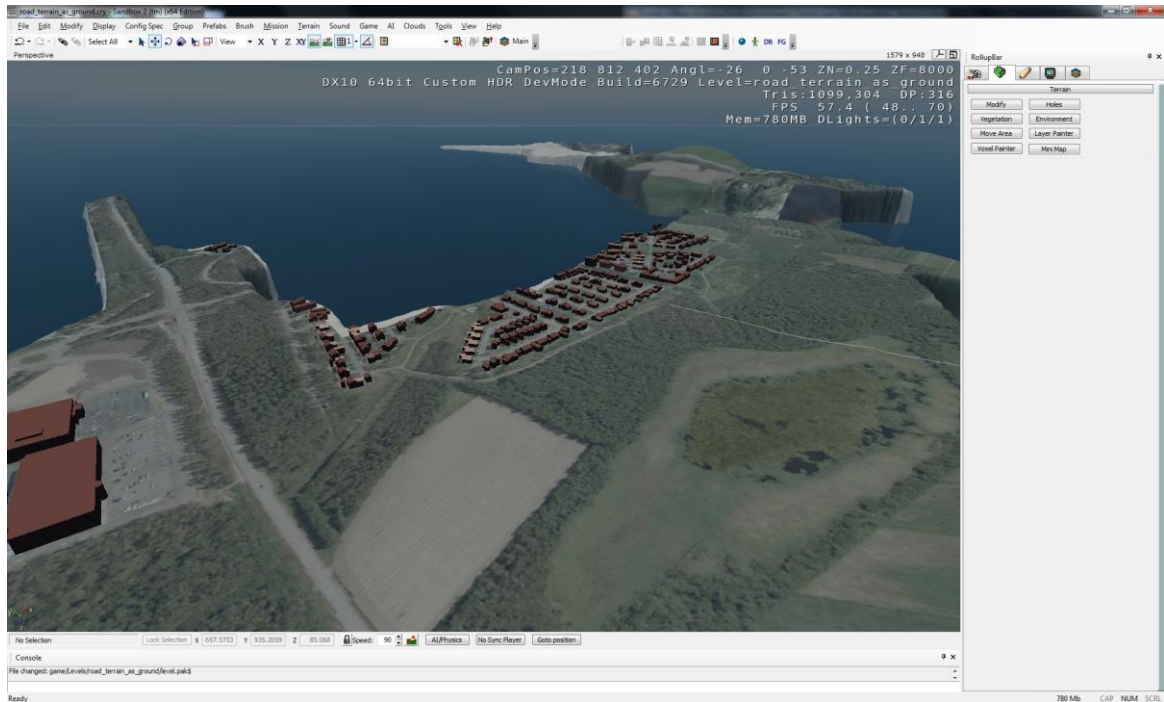
4.1.1 Cryengine 2

Jag började med att bygga upp terrängen och lägga på ortotexturen (se figur 6).
Vattnet var lätt att skapa genom att ange höjden på ytan.



Figur 6: Terräng i Cryengine

Eftersom den nya vägen och bron fanns med i modellen, och då också höjddatan, blev dessa också terräng. Detta kan man justera antingen i höjddatan eller direkt i spelmotorn. Efter jag exporterat geometrin från byggnader, väg och bro kunde jag lägga in dessa i spelmotorn (se figur 7). Det finns inga gemensamma referenspunkter mellan terrängen och modellerna så jag fick manuellt placera ut och justera modellerna.



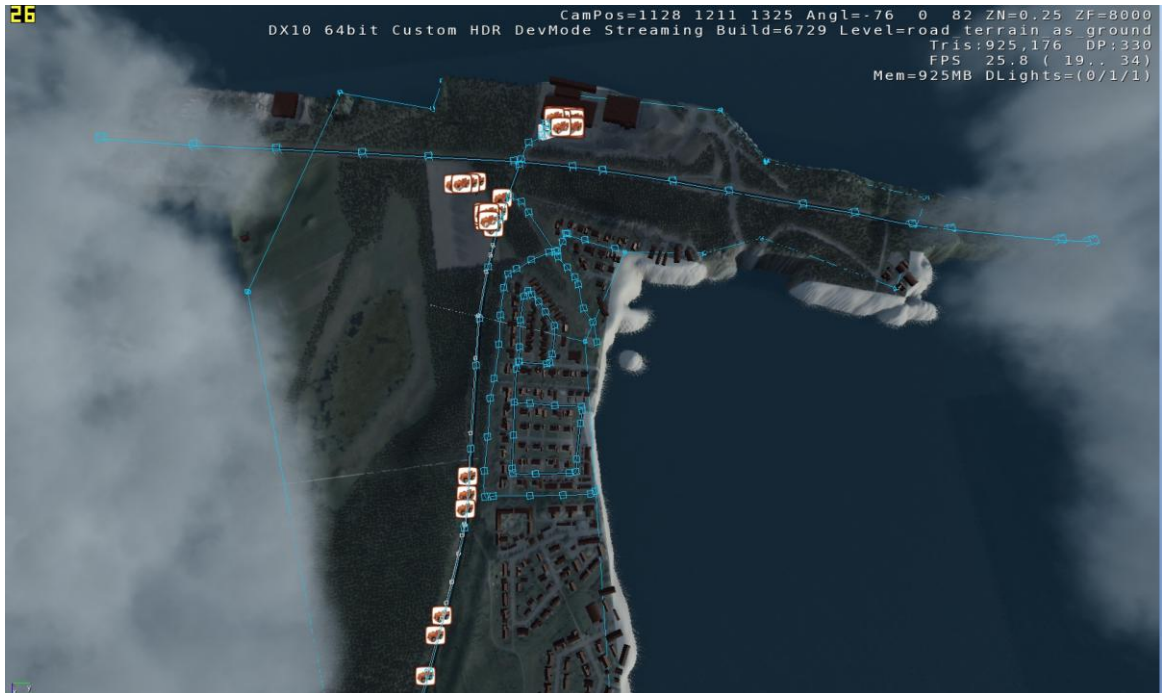
Figur 7: Husmodeller importerade i Cryengine

Det fanns ingen funktion för att placera ut träd automatiskt på maskarna utan det gjorde jag genom att måla ut dem manuellt på rätt plats. Jag skapade även moln som blev animerade då jag lät dem påverkas av vinden (se figur 8).



Figur 8: Träd och moln insatta i Cryengine

För att göra miljön mer verklighetstrogen infogade jag bilar som jag lät köra längs vägarna (se figur 9).



Figur 9: Trafik genererad i Cryengine

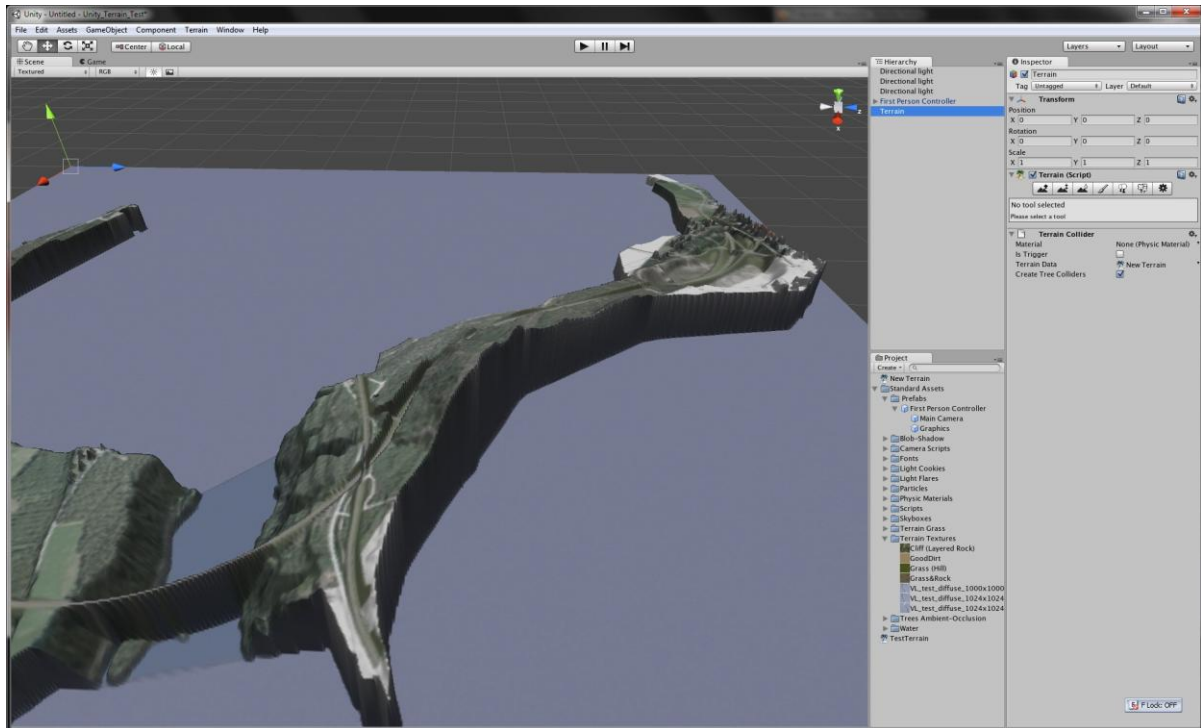
Man kan även som användare köra bilarna, detta gör att man kan få en känsla av hur det kommer vara att köra på den nya vägen (se figur 10).



Figur 10: En användare kör en bil i miljön

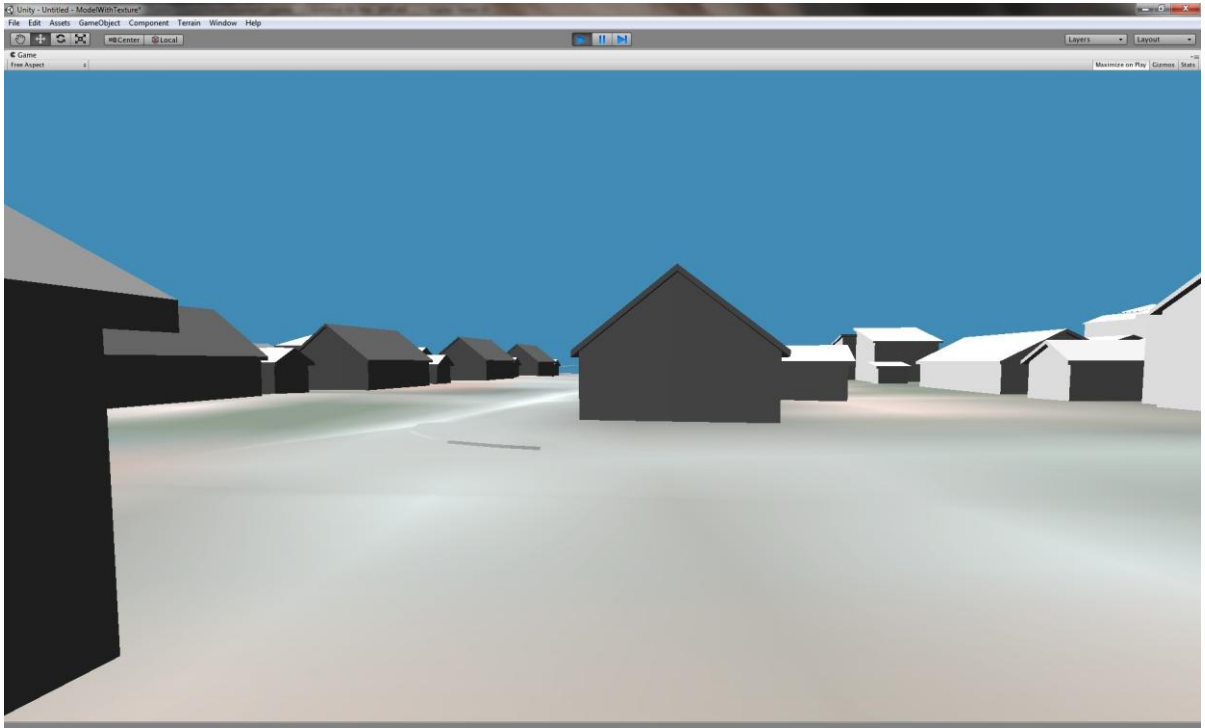
4.1.2 Unity3D

Jag gjorde på liknande sätt i Unity. Jag byggde terrängen med höjddatan och importerade ortofotot (se figur 11).

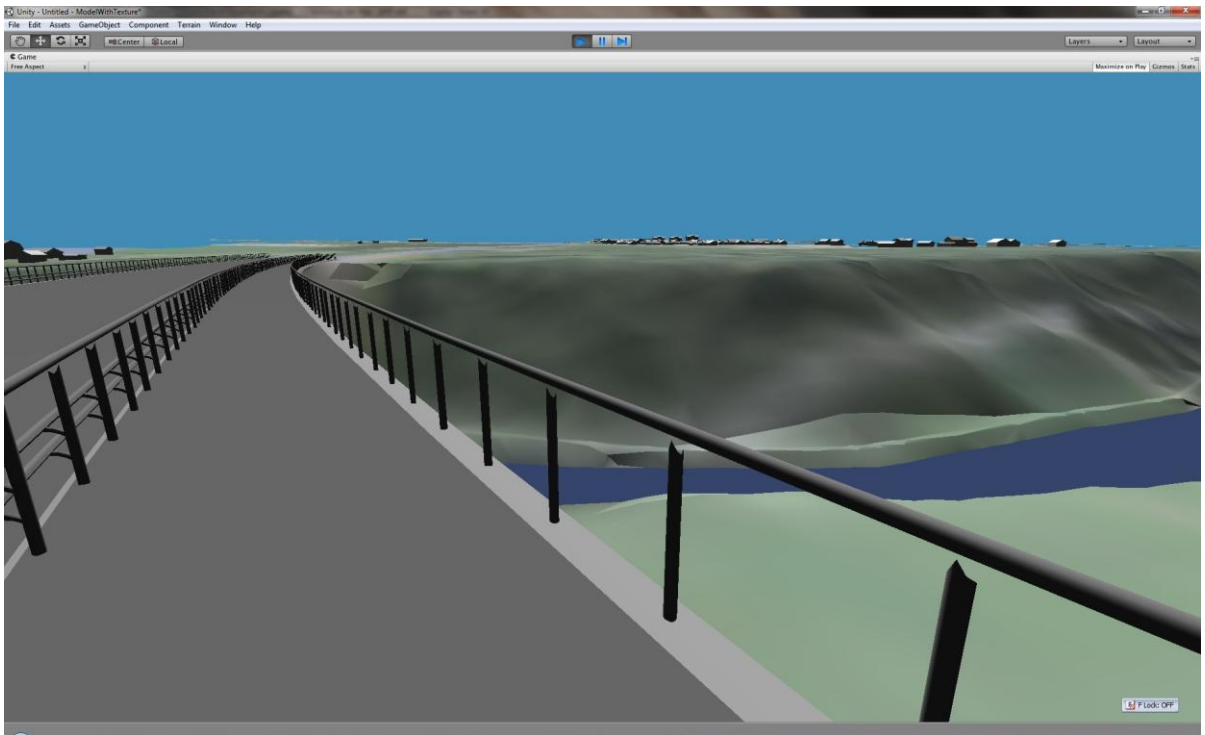


Figur 11: Terräng i Unity

Husen, vägen och bron importerades genom att FBX-filerna placerades i projektkatalogen (se figur 12 och 13).



Figur 12: Hur importerade i Unity



Figur 13: Bromodell importerad i Unity

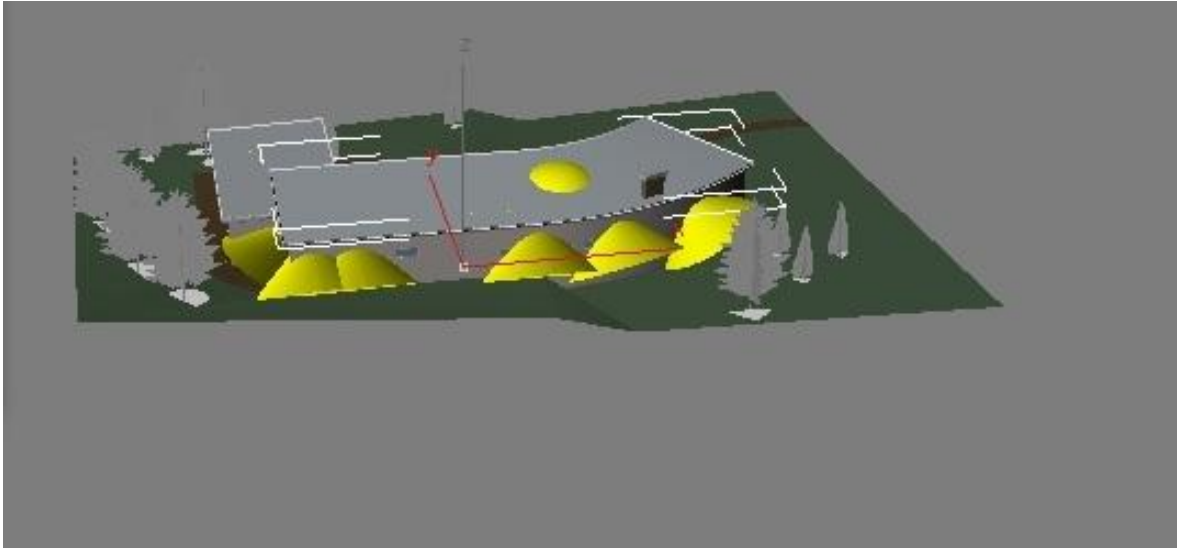
Träden målades på med samma sätt som i Cryengine (se figur 14). Tyvärr finns bara ett fåtal exotiska träd i Unitys bibliotek så det blev inte så naturtroget.



Figur 14: Träd skapade i Unity

4.2 Fallstudie – Hus från Autodesk Revit Architecture

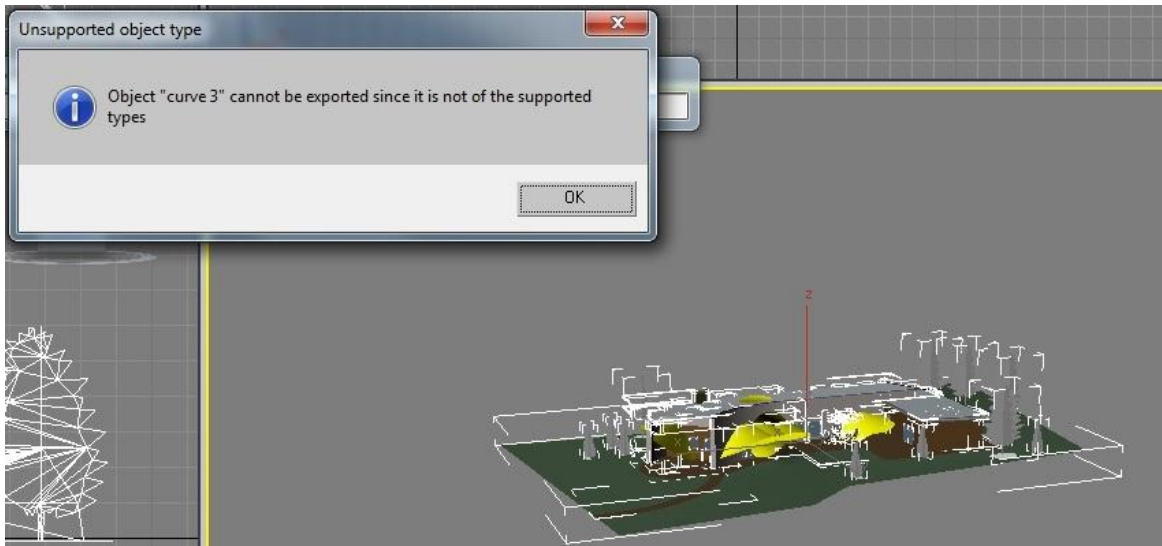
En av verktygen som används för att skapa 3D-modeller är Revit. Jag ville undersöka hur en import från Revit ser ut i spelmotorerna. När jag tog in dem till 3ds Max visade sig att texturerna följde inte med. En lösning jag försökte var att försöka gå genom Autodesk NavisWorks. Då kom texturerna med men ljusen från lamporna blev objekt (se figur 15). Ett annat problem var att man kunde bara se ytor från den ena sidan.



Figur 15: Revitmodell importerad i 3ds Max

4.2.1 Cryengine2

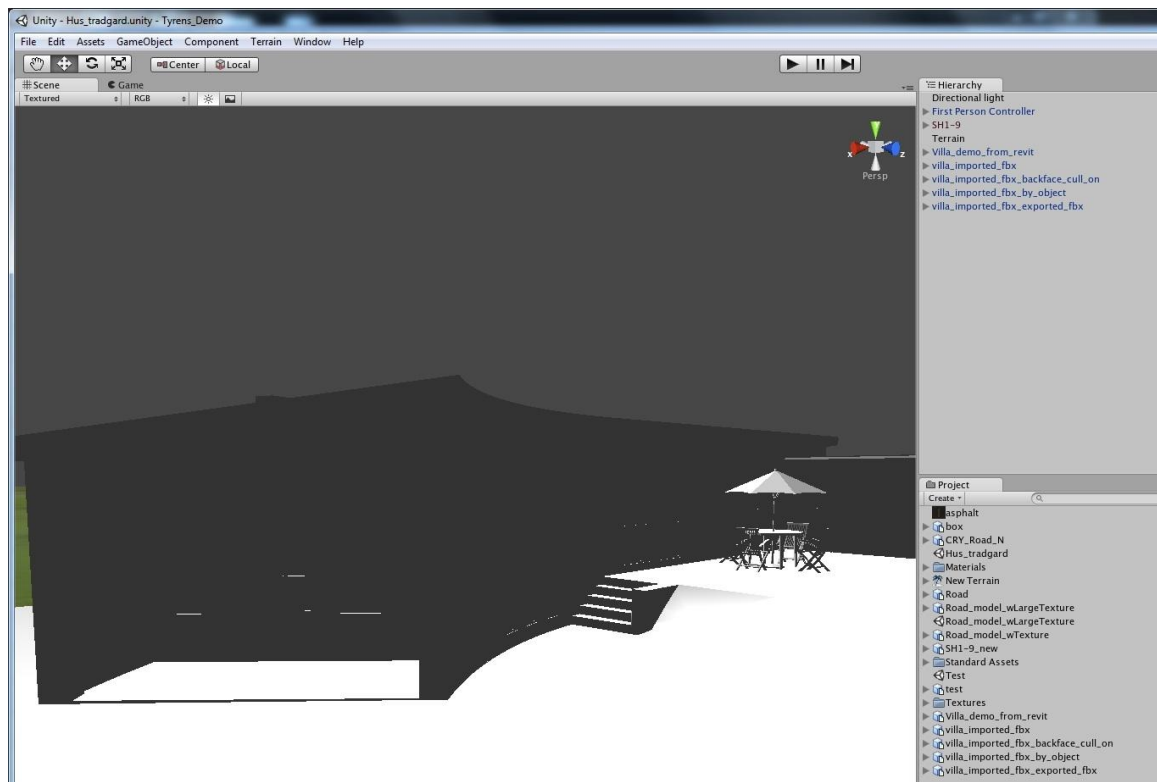
Det gick inte att exportera Revit-modellen till Cryengine på grund av den inte stöder den typen av objekt (se figur 16). En lösning försöktes hittas utan lyckat resultat.



Figur 16: Felmeddelande efter försök att importera till Cryengine

4.2.2 Unity3D

Efter import direkt från Revit till Unity såg modellen likadan ut som i 3ds Max (se figur 17).



Figur 17: Revitmodell importerad till Unity

Även importen från Navisworks utföll som i 3ds Max (se figur 18).



Figur 18: Navisworksmodell importerad till Unity

Unity kollaborerar inte så bra med revitmodeller, i alla fall inte med de modeller jag har använt i mina fallstudier. Programmet kraschar ofta när man jobbar med dessa modeller, speciellt då man flyttar runt objektet. Det tar även lång tid att importera, öppna och skala om jämfört med modeller från andra program.

5 Utvärdering av spelmotorerna

5.1 Licensform och kostnader

5.1.1 Cryengine 2

“Vi licensierar inte CryENGINE® 2 längre.” (Crytek Website - CryENGINE2 Register, 2010). Detta betyder att det är gratis att använda spelmotorn för visualisering inom företaget, bara man inte kopierar och säljer skivor med spelet på. Man måste dock köpa spelet Crysis Wars för att få tillgång till editorn och spelet.

5.1.2 Unity3D

Det finns två versioner av Unity i dagsläget. Den ena är en gratisklient som är helt fri att användas på alla sätt. Den saknar dock en del funktioner som man har tillgång till om man köper den andra mer avancerade versionen, Unity Pro. Den har en engångsavgift \$1500 (US) och då får man uppdatera gratis om nyare versioner kommer ut. Några funktioner som kan vara intressanta från Pro-versionen är:

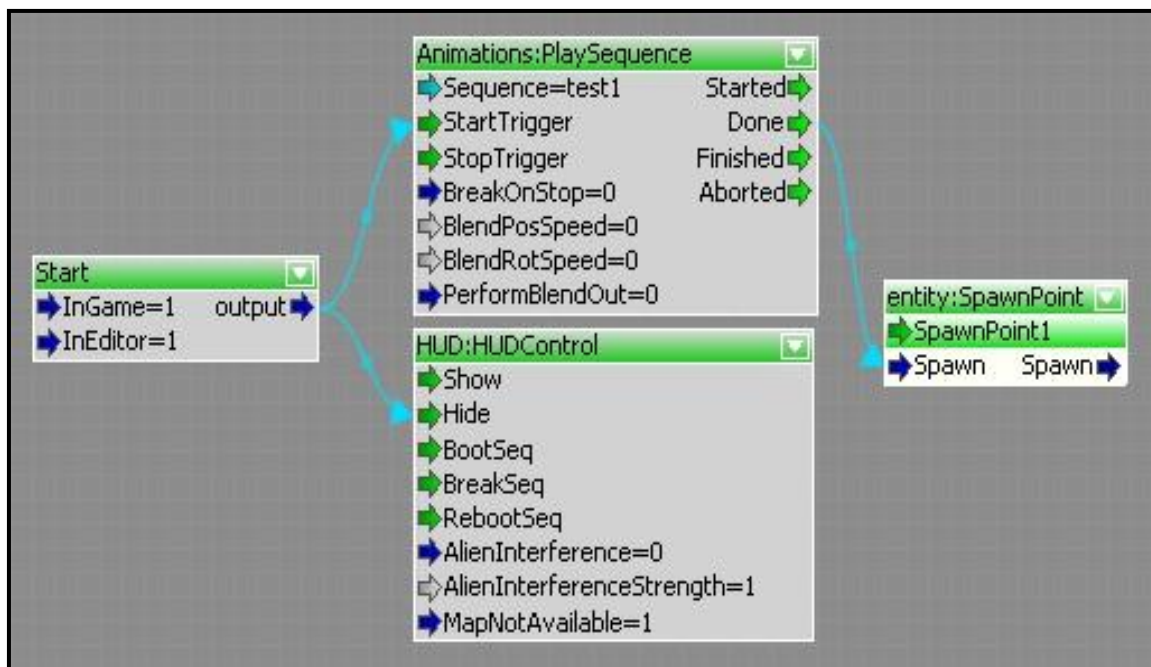
- Massor med positiva grafikförbättringar som t ex Global Illumination och Render-to-Texture Effects för att skapa snyggare vatten.
- Möjlighet att byta ut Unitys splash screen till företagets logga.
- Realtidsskuggor. (Unity3d Website - UNITY: License Comparison, 2010)

5.2 Hur lätt är det att förändra GUI

5.2.1 Cryengine 2

Cryengine har som standardnavigering en avatar med vapen, hälsomätare och en minimap. Detta kan enkelt tas bort med hjälp av flow graphs, som är en grafisk metod att programmera i Cryengine. För att få en snabb inblick i flow graph-programmeringen förklaras här hur man tar bort standardgränssnittet.

I flow graph-vyn skapar man en nod som heter Start, den aktiveras då man startar spelet. Alla noder har en input- och outputsida och man kan dra pilar mellan dessa. För att dölja standardgränssnittet drar man en pil från startnodens output till "Hide" i noden HUDControl (se figur 19). De övriga noderna i figuren aktiverar en animation som följs av en "SpawnPoint", alltså positionen där avataren ska börja spela ifrån.



Figur 19: Flow graph i Cryengine

5.2.2 Unity3D

Att skapa eget GUI kräver en del programmering men möjligheterna är stora. Programmeringsspråket är JavaScript som är en variant av det objektorienterade språket Java. Man kan i princip skapa vilket gränssnitt som helst, inklusive menystrukturer där man kan göra diverse inställningar. Det finns även möjligheter att infoga knappar för navigering till förutbestämda nodpunkter, låta kameran åka med en förutbestämd bana och kunna ta bilder. Unity har ett välutvecklat dokumentationsbibliotek med exempelkod på deras hemsida där man med hjälp av detta kan komma igång snabbt. För att få en känsla av hur det kan se ut infogar jag

ett enkelt exempel på hur man skapar två knappar och hur resultatet ser ut (se figur 20).

```

function OnGUI () {
    // Make a background box
    GUI.Box (Rect (10,10,100,90), "Loader Menu");

    // Make the first button. If it is pressed,
    Application.Loadlevel (1) will be executed
    if (GUI.Button (Rect (20,40,80,20), "Level 1")) {
        Application.LoadLevel (1);
    }

    // Make the second button.
    if (GUI.Button (Rect (20,70,80,20), "Level 2")) {
        Application.LoadLevel (2);
    }
}

```



Figur 10: Knappar i gränssnittet skapade i Unity

5.3 Importering

5.3.1 Cryengine 2

För att kunna exportera geometriska modeller, material och texturer till Cryengine måste man gå genom 3ds Max eftersom det är där man kan installera ett plugin som möjliggör en export till Cryengine.

Eftersom man måste veta hur man skapar Multi/Sub-material krävs det en viss förkunskap i 3ds Max. Det kan även vara bra veta hur man sätter ihop flera objekt till ett stort eftersom det är smidigare att jobba med ett endaste stort objekt i Cryengine jämfört med att ha massor av småobjekt. Ytterligare fördelar har man om man kan

komprimera modeller till färre vertexpunkter då man ibland har en modell med onödigt hög punktupplösning.

I princip allt som kan öppnas i 3ds Max går att importera till motorn, det finns dock begränsningar och regler man måste följa. Modellerna kan inte vara uppbyggd med fler än 65000 punkter. Detta man kan lösa genom att antingen dela upp modellen i lagom stora bitar, eller genom att komprimera modellen. Även texturerna har vissa krav. Sidlängden måste vara en tvåpotens och vara högst 4096x4096. Texturen måste även vara kvadratisk och filformatet TIF och DDS i kombination används endast.

Heightmapen verkar inte ha någon begränsning på storlek utan det är datorprestandan som sätter gränserna. Ett importtest resulterade i att Cryengine kunde utan problem ta in heightmapar med sidlängden 16384x16384. Det betyder att man kan ha en karta på 4096x4096 meter med höjddata på varje 2,5 dm.

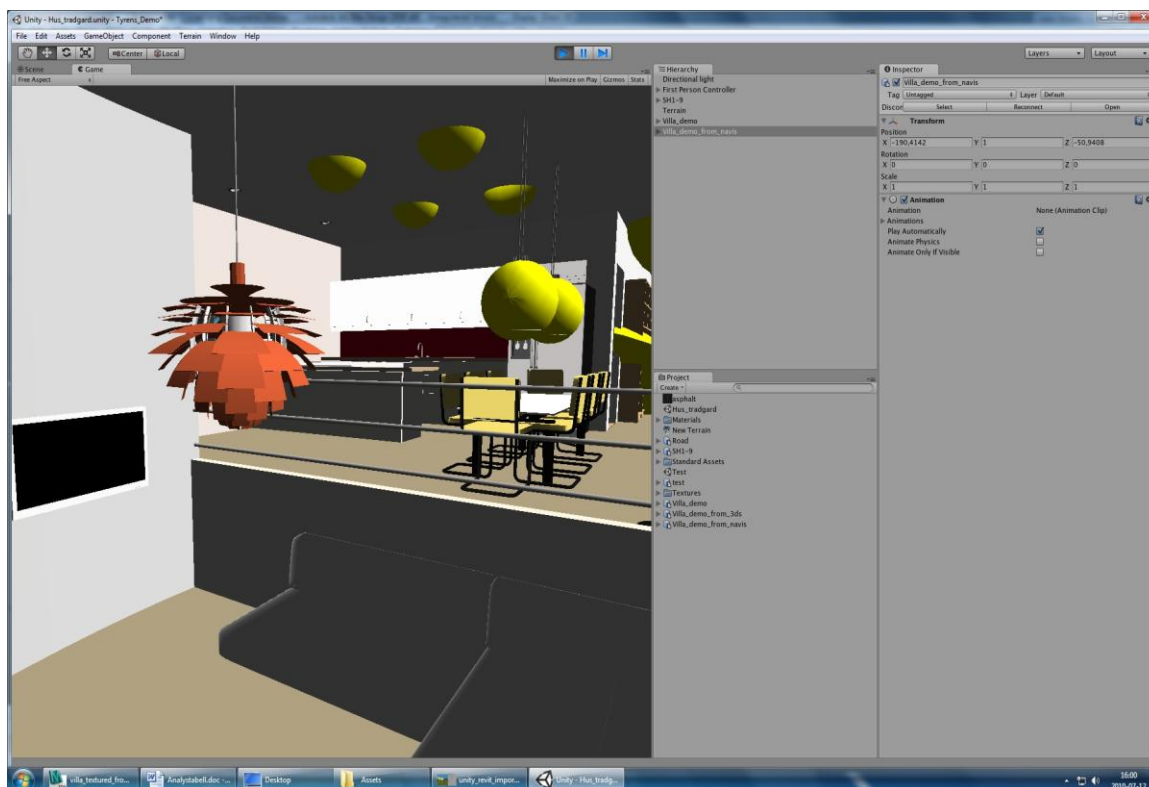
5.3.2 Unity3D

Unity har satsat på att ha bra importmöjligheter och kan importera modeller från de flesta populära 3D-programvaror som t ex:

- Maya
- Cinema 4D
- 3ds Max
- Cheetah3D
- Modo
- Lightwave
- Blender

De filformat som hanteras är bland andra .FBX, .dae, .3DS, .dxf och .obj. Se figur 21 för ett exempel av en FBX-import. För att importera är det lättast att dra sin modell in till Unity och släppa det under projektfliken. Modellerna konverteras då

automatiskt och läggs i projektkatalogen. Man gör likadant med sina texturer. Mappingen på texturerna sparas och den hanterar UVW-map i 3ds Max. Texturerna ska helst vara en kvadrat och Unity hanterar texturen bäst om längden på en sida är en tvåpotens, maximal storlek är 4096x4096. Texturer som inte är i detta format konverteras och skalas till detta format.



Figur 21: En FBX-modell importerad till Unity

Heightmapen måste vara en gråskalebild i filformatet RAW med 16 bitar. Unity klarar inte heightmapar större än 4096x4096 vilket skulle kunna vara ett problem om man vill bygga upp terrängen av ett stort område. (Unity Website - How do I import objects from my 3D app?, 2010)

5.4 Vad följer med vid import

5.4.1 Cryengine 2

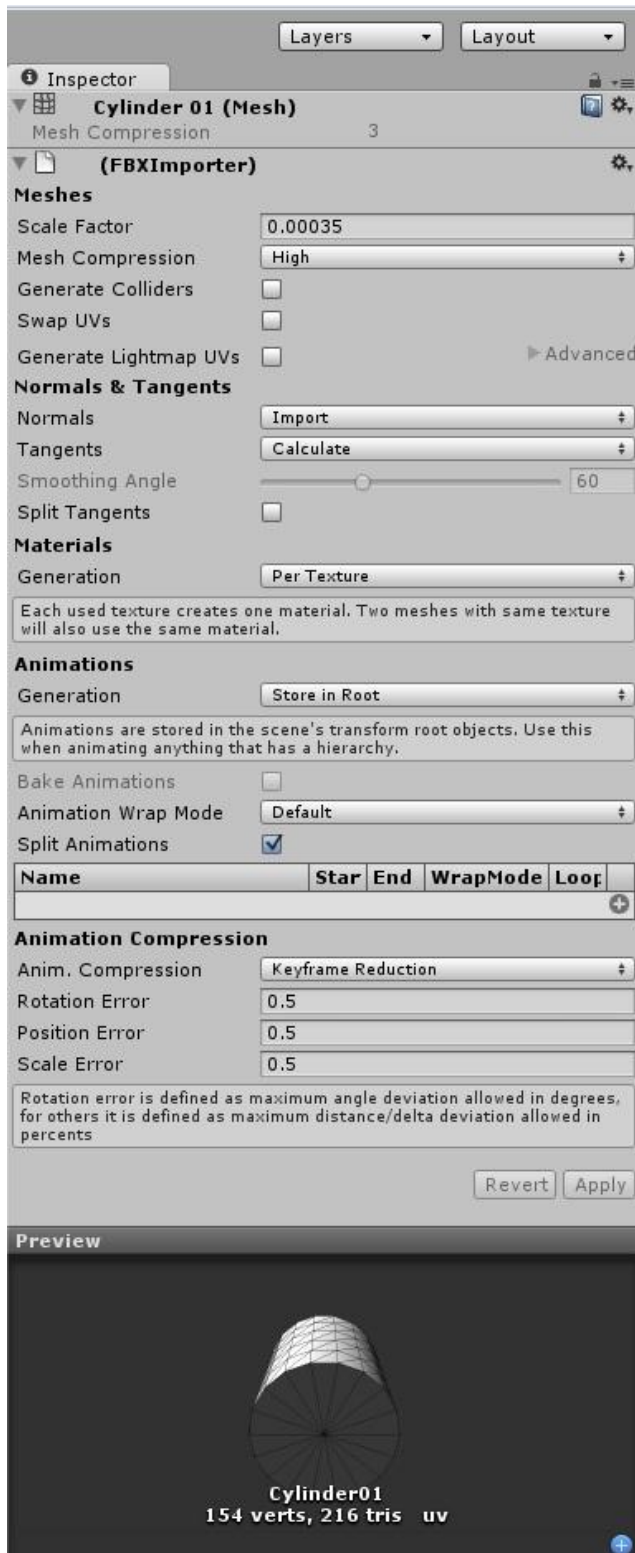
Exporteringen i 3ds Max görs i två steg; först exporterar man geometrin, sedan materialet. Geometrifilen innehåller information om vertexpunkterna, vilket material-ID ytorna har och texturkoordinater. Allt detta sparas som en .cfg-fil. I materialfilen sparas ett multi/sub-material från 3ds Max där alla texturer man använt ligger samlade med olika ID. Man kan även spara olika materialegenskaper som Specular Color och bumpmaps. Effekter som glow, opacity och shader ställer man in i editorn efter man importerat materialfilen.

Exportern klarar ej av så kallade degenerated faces. Detta används ibland för att spara på information, t ex då man ritar en linje och bestämmer en tjocklek istället för att rita en cylinder som har information om varje punkt. Det finns möjlighet att lösa detta i 3ds Max genom diverse inställningar.

5.4.2 Unity3D

Det filformat som har används för import i detta projekt är fbx. Detta är ett filformat som används i de flesta Autodesk-program och är utvecklat för att exportera och importera modeller mellan dessa. I denna fil sparas all information som behövs för att bygga upp modellen i Unity såsom geometrin, texturreferenser och texturkoordinater. Det finns en inbyggd FBX-exporter i Unity som konverterar MAX-filer till FBX då man infogar MAX-filer till Unity (se figur 22.)

Själva texturerna importeras genom att placera bilderna i Unitys projektkatalog. De filformat som stöds för bilder är PSD, TIFF, JPG, TGA, PNG, GIF, BMP, IFF och PICT.



Figur 22: FBX-importer i Unity

5.5 Export av animering/bild

5.5.1 Cryengine 2

Det finns möjlighet att spara skärmdumpar genom att trycka F12. Då sparas en JPG-fil i Cryenginekatalogen. Man kan också ha tidssynkade animeringar genom att rita upp spår i sin spelplan och låta en kamera flyga längs denna. Man skapar lägesnoder där man specificerar position, riktning och brännvidd på kameran. Dessa läggs på en tidsaxel så att man kan ange vilken hastighet allting ska ha. Animeringen kan antingen köras i realtid eller som en filmoutput. Det finns två sätt att spara sin film på. På det ena sättet sparas en JPG-fil för varje frame i en katalog. Man får sedan sätta ihop filmen i ett tredjepartsprogram som gör en filmfil av en samling JPG-bilder. Fördelen med detta sätt är att man kan bestämma filmens FPS, nackdelen är att man inte får med ljudet. Det andra sättet är att installera ett program som heter FRAPS. Den sparar en filmfil i realtid tillsammans med ljudet. Nackdelen är att filmens FPS är beroende av hur snabb datorn kan rendera bilderna.

5.5.2 Unity3D

En skärmdump kan sparas med enkel kodning. Här är ett exempel på hur man sparar en png-bild när man trycker på hoppknappen:

```
if (Input.GetButtonDown("Jump")) // i.e. space{
    Application.CaptureScreenshot("Screenshot.png");
}
```

Även i Unity kan man ha tidssynkade animeringar. Detta skapas på ett liknande sätt som i Cryengine. Dessa kan aktiveras på flera sätt; när man går in i ett område, när man startar spelet eller om man trycker på en knapp som lagts in i gränssnittet. Om man skulle vilja spara en filmfil fungerar FRAPS även med Unity.

5.6 Prestanda/hårdvarukrav

5.6.1 Cryengine 2

Rekommenderade krav:

- Processor: Core 2 Duo/Athlon X2 eller bättre
- RAM: 1.5GB
- Grafikkort: NVIDIA 7800 Series, ATI Radeon 1800 Series eller bättre
- VRAM: 512MB grafikkortsminne
- Hårddiskplats: 12GB
- Ljudkort: DirectX 9.0c kompatibel
- DVD-ROM
- OS: Microsoft Windows XP, Vista, 7
- DirectX: DX9.0c or DX10c. (Crysis Online Website - System Requirements, 2010)

Cryengine kan generera ut stereo men det är bara vissa grafikkort som stöds.

5.6.2 Unity3D

Rekommenderade krav:

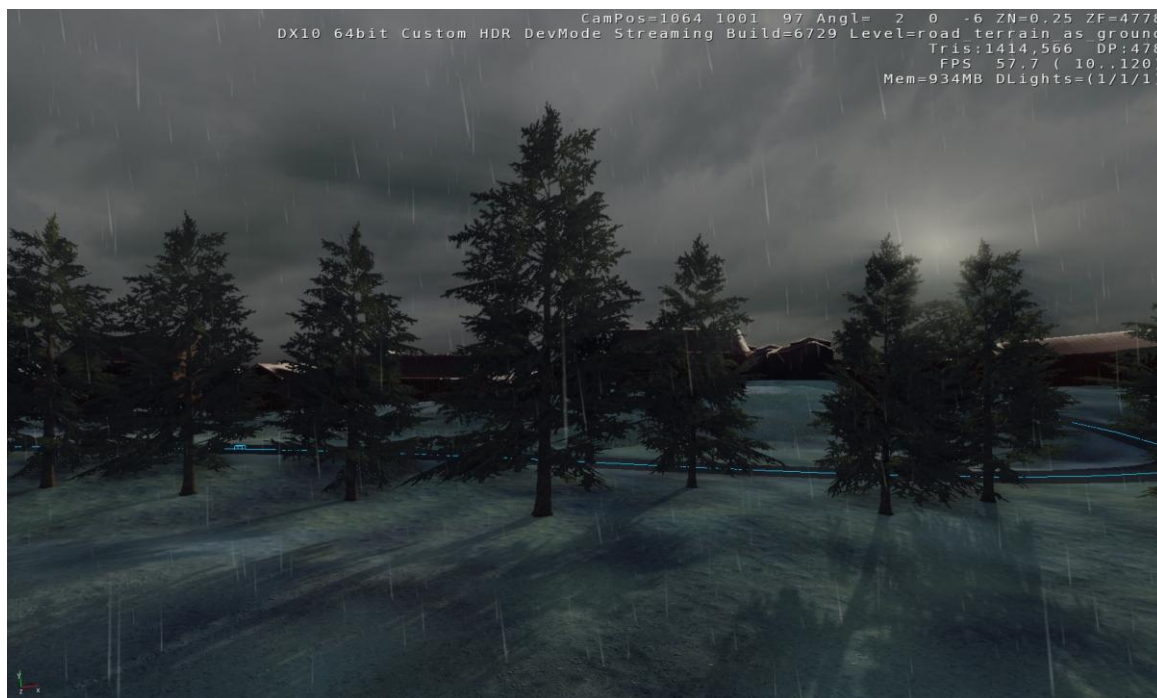
- Windows: XP SP2 eller senare; Mac OS X: Intel CPU & "Leopard" 10.5 eller senare.
- Grafikkort med 64 MB minne. Vilket kort som helst från detta millennium fungerar.
- Resten beror på komplexiteten av projektet. (Unity Website - UNITY: Unity Web Player End User License, 2010)

Unity stöder vissa stereotekniker som t ex Nvidia 3d vision. Det finns även möjlighet att placera två kameror bredvid varandra och skicka ut två olika videosignaler Dessa kan projiceras upp och skapa stereoeffekt.

5.7 Andra användbara funktioner

5.7.1 Cryengine 2

Det finns många möjligheter när det gäller vädersimulering. Ett antal olika sky boxar finns färdiggjorda, allt från solig dag till mulig kväll. Till detta kan man infoga regn- och snöfall som partikeleffekter och även blixtar kan skapas (se figur 23). Vindstyrkan kan ändras, vilket märks på träden, molnen och vågorna på vattnet.

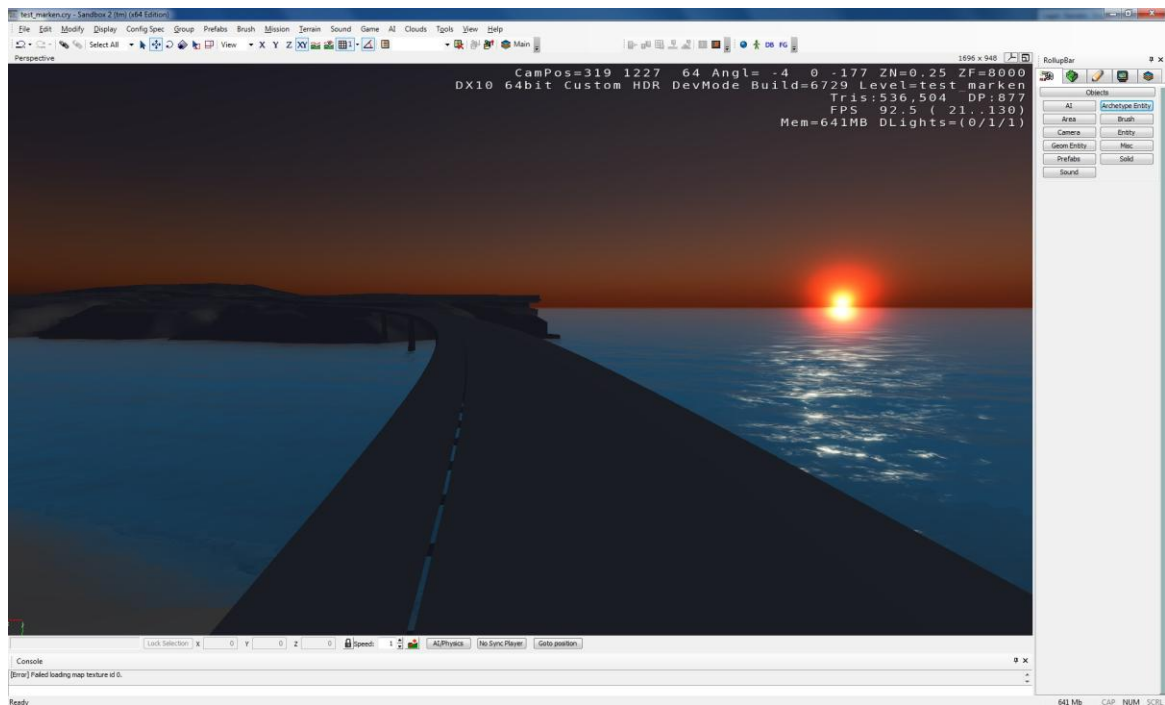


Figur 23: Vädereffekter i Cryengine

Det finns ett stort utbud på färdiggjorda objekt såsom träd, buskar, gräs, människor och fordon. Man kan dessutom ladda ned paket innehållande växtlighet från olika delar av världen. I min demo använde jag ett europapaket med olika sorters tallar och granar. Jag använde även ett antal bilar som kopplades till ett spår på en väg.

Med hjälp av flow graphs kunde jag få dessa att köra längs vägen och på så sätt skapa trafik.

En bra utvecklad funktion är tidssimuleringen. Här finns massor av inställningar som t ex solens bana på himmeln genom att ange vilken bredgrad miljön befinner sig i, vilken tid solen ska gå upp och ner och i vilken riktning i miljön som ska vara norr för att bara nämna några (se figur 24). Med dessa inställningar blir miljön mer verklighetstrogen och kan man då göra skugganalyser.



Figur 24: Solnedgång i Cryengine

Eftersom Cryengine är utvecklat för FPS-spel så är det lätt att öppna sin spelplan i multiplayer-läge. Detta gör att man kan ha guidade visningar där åskarna får en större frihet genom att kunna titta runt och navigera dit de vill. Detta kan göras både genom ett lokalt nätverk och via internet.

De styrenheter som stöds, förutom standardnavigering med tangentbord och mus, är PC-versionerna av Xbox- och Playstationkontrollerna Logitech ChillStream och

Logitech Dual Action. Det finns även ett plugin som gör att man kan använda Nintendo Wii Remote och Nunchuk (Crytek's Official Modding Portal | Mods | Get Your Wiis On!, 2010).

5.7.2 Unity3D

Vädersimuleringen i Unity är inte lika utvecklas som i Cryengine. Himlen blir statisk då man inte kan använda moln med volym. Himlen renderas med hjälp av en sky box som använder en 2D-bild för att måla upp himlen. Det finns mycket begränsat antal objekt att använda med tanke på träd och buskar. Man får då använda andra mjukvaror som kan användas för att skapa växter. Dessa kan sedan importeras till Unity.

Det finns möjligheter att simulera olika tider men det kräver manuellt arbete som t ex att flytta ljuskällan och ändra färgen på ljuset. Detta medför att man inte kan göra tillförlitliga skugganalyser. Unity stöder också multiplayer-läge. Det kräver en hel del kodning där utvecklaren måste ha kunskaper i digital kommunikation och nätverk.

Man styr Unity med hjälp av tangentbord och mus då man kör på PC- och Mac OS-plattformen. Eftersom man kan köra Unity på flera andra plattformar, såsom Wii, Xbox360, Playstation3, Android och iPhone, använder man flera andra styrenheter. Man använder då plattformarnas egna styrenheter, det vill säga handkontroller och pekskärmar.

6 Manual för import

6.1 Cryengine

Denna manual kräver att användaren har goda kunskaper i 3ds Max.

6.1.1 Installation av 3ds Max, motorn och nödvändiga plugins

Operativsystemet jag rekommenderar är Windows 7 64 bitar.

1. Börja med att installera 3ds Max 2009.
2. Installera Crysis och Crysis Wars (Punkbuster och Gamespy ska ej installeras).
3. Installera:
 - a. CrysisWars_patch5.exe
 - b. CrysisWars_ModSDK_SourceCode_v1.0
 - c. CrysisWars_ModSDK_SPMOD_v1.0
 - d. CrysisWars_SDK_Tools_v1.1 (fungerar bara till 3ds Max 2009 och äldre)
4. Gå till "c:\program files (x86)\Electronics Arts\Crytek\Crysis Wars\tools" och starta "SettingsMgr.exe".
5. Om "Engine State" inte är grön, klicka då på knappen "... " och gå till katalogen "c:\program files (x86)\Electronics Arts\Crytek\Crysis Wars" och klicka OK. Då ska "Engine State" bli grön.

6.1.2 Inställningar i 3ds Max efter du installerat motorn

1. Ta fram CryENGINE2 Exporter-knappen
 - a. Under "Utilities"-fliken, klicka på "Configure Button Sets". Lägg till en knapp och scrolla ner i listan till "CryENGINE2 Exporter" och dra den till den tomma knappen.
2. Ställ in rätt enhet
 - a. Gå till Units Setup.
 - b. Välj "Generic Units".

- c. Välj 1,0 Centimeters i System Units Setup.

6.1.3 Exportera material och texturer

1. Om modellen inte är texturerad, hoppa till punkt 2.
 - a. Tryck "Attach List" och markera allt, sedan "Attach".
 - b. Välj "Match Material Ids to Material".
 - c. I Material editorn, klicka på en ny kula och klicka på "Pick material from object" och klicka på din modell. Då skapas en materiallista av typen "Multi/Sub-Object". Hoppa till punkt 4.
2. Skapa ett Multi/Sub-material och lägg in alla texturer som ska användas på objektet.
3. På alla undermaterial, under "Shader Basic Parameter", ändra till "Crytek Shader" och klicka i "Physicalize".
4. Döp ditt huvudmaterial till samma namn som .max-filen.
5. Undersök modellen så alla ytor har rätt material-ID och spara innan du exporterar.
6. Gå till "Utilities"-panelen och klicka på "CryENGINE2 Exporter". Scrolla ner och tryck på "Create Material", då börjar spelmotorn ladda (om den inte redan är öppen).
OBS: Om inte savedialogen inte kommer upp, klicka då en gång till på knappen "Create material" när motorn laddat klart.
7. Skriv in samma namn som din .max-fil och spara till "c:\program files (x86)\Electronics Arts\Crytek\Crysis Wars\Game\Materials".

6.1.4 Exportera modeller från 3ds Max

1. Spara modellen som .max.



Figur 25: Exportplugin

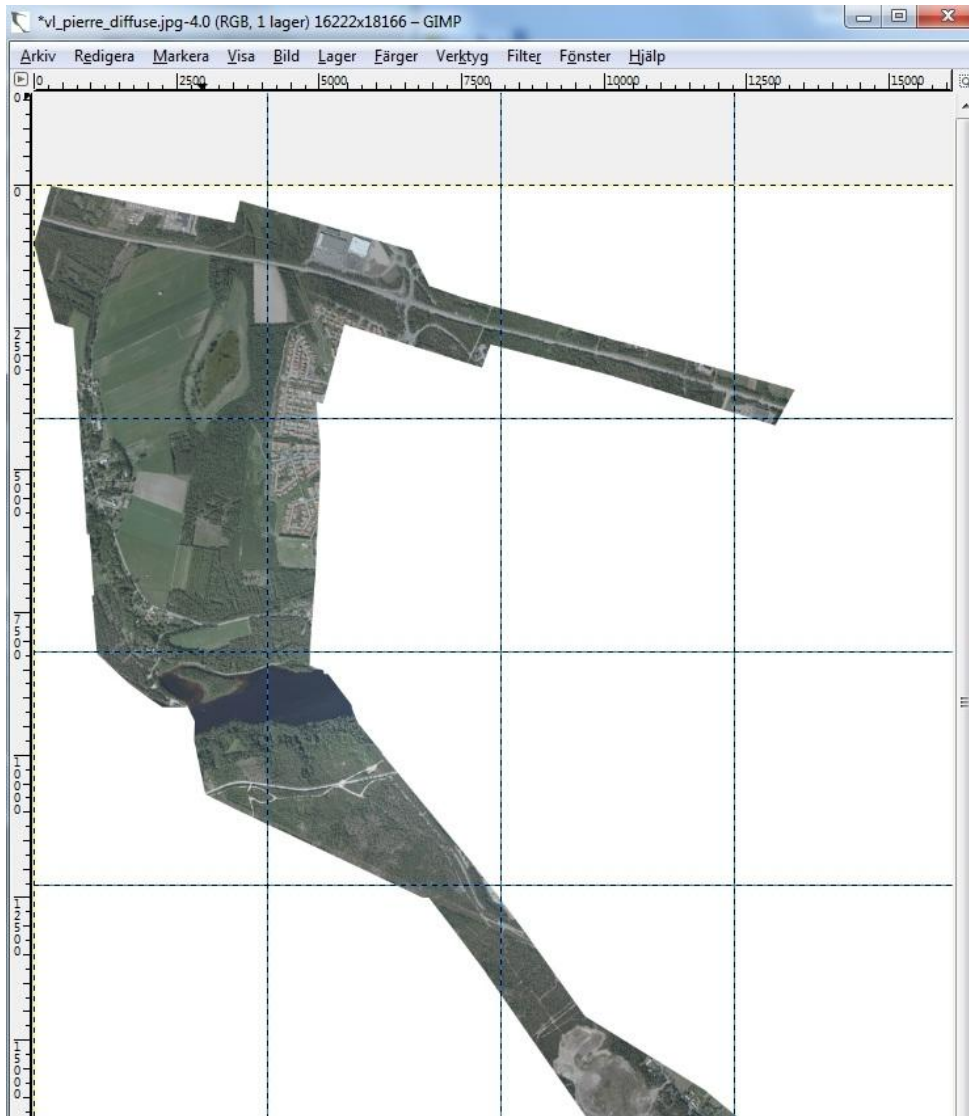
2. Klicka på "Pick..." under CryENGINE2 Exporter och välj din modell (se figur 25.)
3. Exportera 3d-modellen till spelmotorn genom att trycka på "Export Nodes".
4. Ta bort eventuella "Degenerated Faces" genom att trycka på knappen "Select Faces", sedan "OK". Gå sedan in i "modify"-fliken och markera "Faces". Tryck "Delete". Exportera igen så inga fel uppstår.
5. *.cfg-filen skapas där du sparat din max-fil. Flytta cfg-filen till katalogen "c:\program files (x86)\Electronics Arts\Crytek\Crysis Wars\Game\Objects".

6.1.5 Importera modeller till Cryengine

1. Öppna editor.exe som ligger i "c:\program files (x86)\Electronics Arts\Crytek\Crysis Wars\Bin 64 (eller 32)".
2. Du navigerar runt genom att hålla ner höger husknapp, då kan du titta runt med musen och röra dig runt med tangenterna: W=upp, A=vänster, S=ner och D=höger.
3. Tryck på knapparna "Follow terrain" och "Select and move".
4. Tryck på knappen "Brush" och dra in din modell från listan till kartan.
5. Öppna material editorn med knappen "M" och leta upp ditt material. Tryck "assign material to selection".
6. Starta spelet genom att trycka "CTRL" & "G".

6.1.6 Lägga på en högupplöst textur på terrängen

1. Skapa ett flygfoto mitt exempel 16384x16384. Dela upp fotot till 16 st 4096x4096 (kan göras lätt med funktionen guillotine, se figur 26), spara som bmp.
2. I sandbox under Layer Painter tryck Tiles Resolution och ändra alla tiles till 4096x4096.
3. Under terrain->import/export terrain texture markera en ruta och tryck import.
4. Leta upp respektive bild, gör likadant med alla tiles. Tryck generate surface textures.



Figur 26: Guillotinfunktionen

6.2 Unity

Man importerar sin modell genom att lägga in originalfilen, t ex en max-fil, direkt till projektkatalogen. Om man har texturer ska dessa också läggas in. Det har ingen betydelse vart i katalogen man lägger sina filer. (Unity - Importing Objects From 3D Studio Max, 2010)

7 Diskussion

7.1 Cryengine

Cryengine genererar väldigt snygg grafik utan höga krav på hårdvara. Detta med hjälp av att motorn kan t ex lysa upp växter då solen är bakom, projektera skugga från alla objekt med solen som referens och skapa moln som har volym som kan röra sig verklighetstroget i vindens riktning och hastighet.

Programmeringen blir lätt då man använder flow graphs. Man får då en bättre översiktsbild än när man programmerar i text. Något som är positivt är att Cryengine innehåller väldigt många färdiggjorda objekt såsom körbara bilar, böjbara växter och moln.

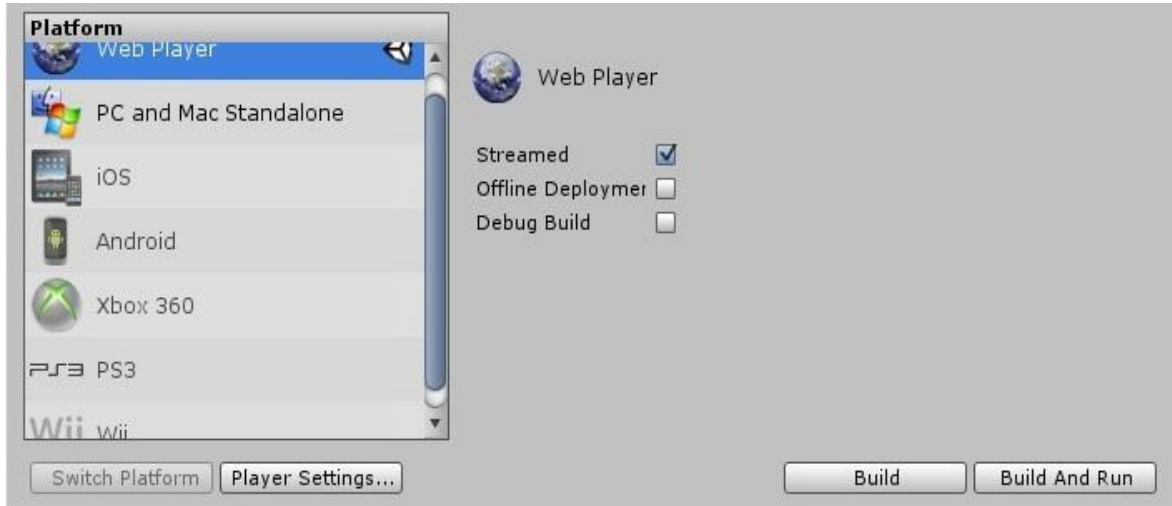
Editorn är den smidigaste av alla spelmotorer jag använt. Man sparar mycket tid genom att man kan ställa in navigationshastigheten. Man kan även spara förinställda positioner och hoppa mellan dessa. Editorn har ett WYSIWYG-gränssnitt, detta sparar också mycket tid genom att man inte behöver hoppa mellan editorn och spelet när man vill testa sin visualiseringskarta.

Möjligheten att kunna använda högupplösta texturer är väldigt fördelaktig, speciellt då man vill projicera ett flygfoto på en terräng.

7.2 Unity

Unitys stora fördel är dess portabilitet. Man kan enkelt spara sitt projekt som en stand alone-applikation så att man inte behöver installera något på datorn för att kunna ta del av visualiseringskartan. Då skapas en katalog med alla nödvändiga resurser och en exekverbar fil som startar upp applikationen. Man kan även skapa en "web player" som kan integreras i en webbplats. Då kan man köra kartan direkt i sin webbläsare och alla med internetuppkoppling kan ta del av projektet. Unity är även portabel på det sätt att man kan köra Unity på flera olika plattformar. De plattformar som stöds är

Windows, Mac OS, Android, iPhone, Wii, Xbox360 och Playstation 3 (se figur 27). Detta gör att man kan nå ut till en större publik och använda styrenheter som kan vara lättare att använda.



Figur 27: Plattformer som stöds av Unity

Unity har satsat på mycket på att vara kompatibel med flera olika mjukvaror. Den har inbyggda importörer och konverterare så att man inte behöver installera plugins eller konverteringsverktyg. Detta är en stor fördel eftersom modellerna som ska användas kan komma från olika modelleringsprogram.

8 Slutsats

Man kan inte importera alla sorters modeller, det finns en hel del att tänka på som t ex backfacing, antal punkter och trianglar i modellen, enheter och skala, referenspunkter, material och texturer.

Båda spelmotorerna har sina för- och nackdelar. Vill man ha en mer uppslukande virtuell verklighet där miljön är mer dynamisk är Cryengine bäst lämpad (se figur 28). Solen, molnen och alla växter rör sig. Flygfotot har en mycket högre upplösning på marken och tiden kan ändras.



Figur 28: Cryengines grafik

Vill man ha en mer distribuerbar spelkarta där sitt projekt kan skickas till kund med e-post eller läggas upp på en webbplats är det Unity man ska satsa på. Den är även mer anpassningsbar och kan utformas och paketeras precis som man vill.

9 Käll- och litteraturförteckning

Crysis Online Website - System Requirements. (den 20 10 2010). Hämtat från
<http://www.crysis-online.com/page/system-requirements.html>

Crytek Website - CryENGINE2 Register. (den 20 10 2010). Hämtat från
<http://www.cryengine2.com/register>

Crytek's Official Modding Portal | Mods | Get Your Wiisis On! (den 20 10 2010). Hämtat från
<http://www.crymod.com/thread.php?threadid=21042>

Tyréns Website - Om Tyréns. (den 20 10 2010). Hämtat från
<http://tyrens.se/sv/Toppmeny/Om-Tyrens/>

Unity - Importing Objects From 3D Studio Max. (den 27 10 2010). Hämtat från
<http://unity3d.com/support/documentation/Manual/HOWTO-ImportObjectMax.html>

Unity Website - How do I import objects from my 3D app? (den 31 10 2010). Hämtat från
<http://unity3d.com/support/documentation/Manual/HOWTO-importObject.html>

Unity Website - UNITY: Unity Web Player End User License. (den 20 10 2010). Hämtat från
<http://unity3d.com/unity/system-requirements.html>

Unity3d Website - UNITY: License Comparison. (den 20 10 2010). Hämtat från
<http://unity3d.com/unity/licenses>