

Jämförelse av 3D-spelmotorer

Marco Lindgren

36647

Kandidatavhandling i datateknik

Fakulteten för naturvetenskaper och teknik

Åbo Akademi

Handledare: Kristian Nybom

3.4.2017

Referat

Under de senaste årtiondena har spelmotorer dykt upp inom spelindustrin. De erbjuder på många olika möjligheter för spelutvecklare. I avhandlingen introduceras först konceptet om spelmotorer och efter det väljs och presenteras tre spelmotorer: Unity 5, Unreal Engine 4 och CryEngine V. Till slut jämförs de med varandra. Resultatet av jämförelsen samlas ihop i en tabell, eller en jämförelsematrix, vars avsikt är att klart lägga fram rå data om spelmotorerna i fråga.

Resultatet av jämförelsen visar att varje spelmotor har sina starka och svaga sidor. Vilken spelmotor som utvecklaren bör välja beror nästan fullständigt på spelprojektets krav på prestanda, pris, m.m.

Nyckelord: spelmotor, jämförelse, Unity, Unreal Engine, CryEngine

Innehållsförteckning

| | |
|---|----|
| Referat | 1 |
| 1. Inledning | 3 |
| 2. Bakgrund | 4 |
| 2.1. Vad är en spelmotor? | 4 |
| 2.2. Utvecklingen av spelmotorer | 5 |
| 3. 3D-spelmotorer | 6 |
| 3.1. Unity 5 | 6 |
| 3.2. Unreal Engine 4 | 7 |
| 3.3. CryEngine V | 9 |
| 4. Konstruktion av en jämförelsematrix | 10 |
| 4.1. Definiering av jämförelsekriterier | 10 |
| 4.2. Jämförelse | 11 |
| 4.1.1. Jämförelse av inbyggda funktioner | 11 |
| 4.1.2. Jämförelse av användarupplevelse | 13 |
| 4.1.3. Jämförelse av utvecklings- och exportplattformar | 14 |
| 4.1.4. Jämförelse av pris | 16 |
| 4.3. Resultat | 18 |
| 5. Avslutande diskussion | 20 |
| Referenser | 21 |

1. Inledning

Under de senaste 30 åren har spelmarknaden växt till en enorm industri. På början av 1990-talet uppfanns konceptet av spelmotorer, och idag drivs de flesta spelen av en spelmotor av något slag. Fastän några spelföretag väljer att utveckla sina motorer enbart för egen användning finns det också en del studior som säljer licenser för de spelmotorer de har utvecklat.

Eftersom det idag finns så många spelmotorer blir det utmanande för en spelutvecklare att välja den mest lämpliga spelmotorn till dess projekt. De mest kända spelmotorerna är i synnerhet stort sett ganska likadana, men de kan ha faktorer som gör den ena mer lönande än den andra.

Syftet med avhandlingen är att göra en klar jämförelse av välkända 3D-spelmotorer samt ge en kort översikt av dem. Detta åstadkoms med hjälp av en jämförelsematris, d.v.s. en tabell som lägger ut rå fakta om de valda spelmotorerna. Informationen som används baserar sig huvudsakligen på spelmotorernas dokumentationer, men också böcker om spelmotorer, konferenser och intervjuer med utvecklare.

Spelmotorerna som behandlas i avhandlingen är avgränsade till de kända 3D-spelmotorerna Unity 5, Unreal Engine 4 och CryEngine V. Unity och Unreal Engine valdes för att de är mycket populära bland hobbyutvecklare, medan CryEngine inkluderades för att den inte längre är populär bland varken stora eller små spelutvecklare fastän den är tekniskt avancerad. Jämförelseområdena fokuserar huvudsakligen på inbyggda funktioner, användarupplevelse, stödda plattformar och pris. Jämförelsen går inte in på djupa detaljer, utan håller innehållet relativt kompakt. Före själva jämförelsen börjar definieras termen "spelmotor" och historien bakom dem presenteras. Efter det lyfts en vald spelmotor i taget fram för att introducera dem till läsaren.

2. Bakgrund

2.1. Vad är en spelmotor?

På en allmän nivå kan en spelmotor beskrivas som en samling återanvändbara mjukvarukomponenter eller "sub-motorer". Till dessa sub-motorer hör oftast en grafikrenderingsmotor som bestämmer vad som ritas på skärmen och en ljudmotor som hanterar ljudeffekter och musik. Ytterligare består de flesta moderna spelmotorer av moduler som ansvarar för minneshantering, artificiell intelligens och nätverkskommunikation. Zerbst [1] konstaterar att en sub-motor måste kunna hantera all data inom sitt ansvarsområde, beräkna all data enligt uppgiftsområdets krav, vid behov föra vidare all data till nästa programinstans samt ta emot data från föregående instans och hantera dessa och utföra beräkningar. Han påpekar också att många spelutvecklare helt enkelt inte bryr sig om hur de underliggande funktionerna är implementerade, så länge de fungerar som förväntat.

Spelmotorer underlättar spelutvecklingen genom att erbjuda samma grundläggande teknologier för flera olika projekt. Detta sparar både tid och pengar eftersom utvecklarna inte behöver programmera ett spel från början, utan istället kan koncentrera sig på att förverkliga sina planer. De flesta moderna spelmotorer stöder även enkel anpassning av spelen till ett antal olika plattformar som t.ex. datorer, spelkonsoler och mobiltelefoner [2][3]. Detta underlättar spelutvecklarnas arbete och tillåter dem att distribuera sina produkter till så många konsumenter som möjligt.

Gregory [4] påpekar att enskilda spelmotorer oftast designas med tanke på ett litet antal olika spelgenrer. Kraven på spelmotorn skiljer sig avsevärt om slutprodukten t.ex. skall vara ett skjutspel med förstapersonsperspektiv eller ett realtidsstrategispel. Ett skjutspel måste klara av att rendera stora, detaljerade tredimensionella miljöer effektivt och snabbt, medan ett strategispel istället siktar på att rita många olika trupper på skärmen samtidigt och att den artificiella intelligensen tänker så strategiskt som möjligt inom spelets regler. Gregory lägger dock till att de populäraste spelmotorerna idag är mycket flexibla och kan användas till genrer och ändamål de inte var gjorda för.

Under det senaste årtiondet har spelmotorernas teknik mognat så mycket att de också blivit goda alternativ för animerad filmproduktion och datavisualisering. Eftersom de flesta populära spelmotorer innehåller sofistikerade fysikmotorer behöver en filmproducent inte animera allt, utan kan låta programmet simulera rörelser och utföra miljöanimationer istället [5]. Friese et al. [6] använde ett antal olika spelmotorer för att visualisera laserskannade grottor från ett förstapersonsperspektiv och kom fram till att spelmotorer har en mycket hög potential för datavisualisering. Slutsatsen stämmer överens med Fritschs och Kadas [7] resultat i ett liknande experiment.

2.2. Utvecklingen av spelmotorer

Stanton [8] berättar att datorspelens ursprung kan spåras tillbaka till år 1951 när Christopher Strachey skapade *Draughts*, en digital anpassning av damspelet. Fastän datorer redan tidigare hade använts för matematiska spel var *Draughts* den första som hade grafiker på en skärm. Det var ändå inte förrän början på 1970-talet som spelindustrin fick sin verkliga början i formen av textbaserade, interaktiva program. Sedan dess har datorspelen utsatts för stora förändringar och har blivit allt mer avancerade både grafiskt och tekniskt.

Fram till 1990-talet var koden i datorspelen i stora drag skrivna för engångsanvändning. Eftersom systemresurserna var mycket begränsade var det viktigt att optimera koden så bra som möjligt för den avsedda plattformen. Allt detta ändrade när spelmotorerna introducerades till världen.

Enligt Gregory [4] och Lowood [9] populariserades termen "spelmotor" år 1993 i samband med spelet *Doom*. Företaget id Software designade spelet så att grafikrenderingen, ljuduppspelningen och fysiksimuleringen var separerade från grafikfilerna, ljudfilerna och banorna. Tack vare denna design var det mycket lätt att modifiera och lägga till resurser utan att behöva röra spelets kärna, d.v.s. spelmotorn. *Doom* blev en stor succé och ett antal spelutvecklare och amatörer fick licens att använda dess motor, "Doom Engine". Detta gav samtidigt upphov till den populära modifikationsscenen.

Lowood [9] lägger dock till att fastän id Software inte talade offentligt om spelmotorer förrän 1993 hade företaget uppfunnit konceptet redan två år tidigare. Företagets skickligaste programmerare John Carmack experimenterade med 3D-grafikrendering som skulle användas för spelserien *Commander Keen*. Carmack och hans kollega John Romero började kalla den gemensamma programdelen "Keen Engine". Eftersom de båda männen var bilentusiaster bestämde de sig för att använda ordet "engine" i namnet. De ansåg att bilens motor liknade ett hjärta och tyckte att det återanvändbara programmet var jämförbart.

Efter *Dooms* stora popularitet blev konceptet av spelmotorer allt mer attraktivt för spelindustrin. På slutet av 1990-talet designades många spel med tanke på modifikation och återanvändning, t.ex. id Softwares *Quake III Arena* och Epic Games *Unreal* [4]. Dessa spelmotorer var lätt modifierbara eftersom de inkluderade egna skriptspråk som kunde användas till att ändra på allt från spelets regler till det grafiska utseendet.

Sedan början av 2000-talet har spelmotorer blivit allt vanligare. Nämnvärda spelmotorer från det årtiondet är Source, RAGE, id Tech 4 och Gamebryo. Under 2010-talet har det också blivit populärt att göra spelmotorerna tillgängliga för nybörjare och små spelutvecklare. På grund av det är t.ex. de senaste versionerna av de populära spelmotorerna Unity och Unreal Engine gratis för amatörer.

3. 3D-spelmotorer

För en spelutvecklare finns det idag en massa olika spelmotorer att välja mellan. Skribenten har valt att jämföra de senaste versionerna av de tre mest kända spelmotorerna. Dessa är Unity 5, Unreal Engine 4 och CryEngine V. Avsikten med detta kapitel är att ge en översikt för de enskilda motorens historik samt en beskrivning över dess basfunktionaliteter.

3.1. Unity 5

Utvecklingen av Unity påbörjades år 2003 av Unity Technologies som bestod av de tre danska programmerarna David Helgason, Nicholas Francis och Joachim Ante [10]. Den

första versionen fick namnet Unity3D och lanserades år 2005 på teknologiföretaget Apples Worldwide Developers Conference -tillfälle. Avsikten var att erbjuda spelutvecklarna en rimligt prissatt professionell spelmotor i stil med hur Apple marknadsförde sin Final Cut Pro-produkt för filmskapare [11]. Den nyaste versionen, Unity 5, är gratis så länge utvecklaren förtjänar under \$ 100 000 per år. Om inkomsterna är högre blir utvecklaren tvungen att köpa en licens som kostar från 32 € per månad till 115 € per månad [12]. På grund av användarvänligheten är Unity en populär spelmotor både bland amatörer och stora spelföretag.

Det första operativsystemet som Unity3D stödde var Apples Mac. Detta visade sig vara en dålig idé från ett affärsperspektiv eftersom plattformens marknadsandel inom spelindustrin var obetydlig. Kort efter släppte de ut en Windows-version samt stöd för webbläsare [11]. År 2008 läggde de till stöd för mobiltelefonen iPhone efter att Apple avslöjade sin appbutik till plattformen. Unity var den första spelmotorn som stödde iOS-operativsystemet och som ett resultat blev efterfrågan mycket hög och Unity Technologies blev strax efter en av de snabbast växande företagen inom industrin. Idag stöder Unity över 20 olika plattformar, bl.a. Linux, Android, Xbox One, PlayStation 4, Nintendo Switch, samt ett antal olika VR- och AR-glasögon [2].

Unitys underliggande sub-motorer är programmerade i C++ medan användargränssnittet är gjort i Mono. Spelutvecklarna kan använda antingen JavaScript eller C# i sina Unity-projekt [11]. Förut var också Boo ett alternativ, men språket stöds inte mer p.g.a. att för liten andel av programmerarna använde det.

3.2. Unreal Engine 4

År 1998 utvecklade Epic Games förstapersonskjutaren *Unreal* vars syfte var att tävla med id Softwares *Quake*-serie. Motorn som de utvecklade för spelet hanterade allt från 3D-rendering och kollisionsdetektion till artificiella intelligenser och nätverkskommunikation [13]. Efter att spelföretagen Legend Entertainment och Microprose visade intresse för *Unreals* underliggande teknologi gjorde Epic Games beslutet att licensiera spelmotorn till andra

utvecklare under namnet "Unreal Engine" [14]. Den första Unreal Engine -spelmotorn användes totalt i ett tiotal olika spel varav de flesta var förstapersonskjutare.

På början av 2000-talet blev spelkonsoler allt mer populära. Fastän Unreal Engine officiellt endast stödde PC-plattformar hade några spelutvecklare anpassat spel gjorda med den till Xbox och PlayStation 2. För att göra anpassningsprocessen smidigare utvecklade Epic Games sin spelmotor vidare och strax efter lanserade de Unreal Engine 2 [14]. Den nya motorn blev populärare än en enda föregående spelmotor. Den användes totalt i över 190 spel. Unreal Engine 2 var också en av de första spelmotorerna som inte var strikt bunden till en viss genre, utan kunde användas till en massa olika stilar av spel.

När HD-konsolerna Xbox 360 och PlayStation 3 anlände på marknaden var Epic Games igen tvungna att uppgradera sin spelmotor så att den kunde möta den nya generationens grafiska krav [14]. De inkluderade ett antal nya belysnings- och skuggsystem i programmet och släppte ut det år 2007 med namnet Unreal Engine 3 [15]. Ett av målen med Unreal Engine 3 var att göra spelmotorns grafiska gränssnitt så användarvänligt att spelets designers och konstnärer kunde använda det för sina ändamål utan hjälp av en programmerare. På grund av att den var så lättanvänd blev spelmotorn en stor succé och idag har den använts i över 400 olika spel av både mindre spelutvecklarstudior och stora företag. Fastän PC, Xbox 360 och PlayStation 3 var de enda plattformarna som spelmotorn först stödde var det senare även möjligt att anpassa spelen till iOS och Android-mobiloperativsystemen [16]. År 2009 lanserades en gratisversion av Unreal Engine 3 kallad "Unreal Development Kit" som var riktat till allmänheten [17].

De nästa utmaningarna för Epic Games var att fortsätta driva spelgrafiken framåt, göra användarupplevelsen ännu smidigare och att göra spelmotorn så skalerbar som möjligt [18]. En helt ny belysningsmotor introducerades för att skapa effekter i realtid som tidigare krävde statiska ljuskällor. Detta tillsammans med nya partikelmotorer, animationsmotorer och post-process effekter gjorde det möjligt för dem att relativt lätt skapa realistiska omgivningar och modeller. Användarupplevelsen förbättrades så att kompileringstiden förkortades till en bråkdel av den gamla. Det var nu också möjligt att interagera med spelvärlden medan man modifierade variabelvärden och funktioner, vilket gjorde prototyperingsprocessen mycket

smidigare [19]. Skaleringsförbättringar inkluderades för att underlätta plattformsanpassning. Allt detta inkluderades i slutprodukten som nu hette Unreal Engine 4 och lanserades år 2014.

Fastän Epic Games först krävde att alla användare av Unreal Engine 4 skulle betala en licensavgift ändrade de affärsmodellen ett år senare så att spelmotorn var gratis åt alla, men att 5 % av slutproduktens inkomster skulle betalas åt Epic Games [20]. Detta tilltalade speciellt mindre studior som nu hade möjligheten att använda professionella utvecklingsverktyg utan att behöva investera en stor del pengar på dem.

Unreal Engine 4 är skrivet i C++. Användarna kan välja mellan att programmera i C++ eller med att använda "Blueprints Visual Scripting", ett programmerbart flödesschema som främst hjälper icke-programmerare skapa funktioner och visualisera vad som händer i koden [21]. I gamla versioner av Unreal Engine var det också möjligt att programmera i UnrealScript, ett språk som påminner om Java [22].

3.3. CryEngine V

Cryteks CryEngine uppkom först i en teknisk demonstration med namnet *X-Isle: Dinosaur Island* på elektronikmässan E3 i Los Angeles år 1999. Demons detaljerade 3D-grafiker imponerade grafikortstillverkaren Nvidia [23], och två år senare var den inkluderad med all Nvidia GeForce 3 -grafikkort som ett verktyg för prestandamätning [24]. Kort efter visade spelförlaget Ubisoft intresse i *X-Isle* och finansierade ett fullständigt spel som skulle baseras på demons teknologier. Det resulterade spelet *Far Cry* lanserades år 2004.

Crytek blev snabbt kända för att skapa spel med mycket höga prestationskrav. Detta rykte förstärktes år 2007 i samband med att spelet *Crysis*, som använde den nya spelmotorn CryEngine 2, krävde en dator som var mycket kraftigare än den tidens normala speldatorer. Detta resulterade i att varje gång ett nytt grafikort lanserades uppkom frågan "But can it run *Crysis?*", eller "Kan den köra *Crysis?*" [25]. I detta skede var CryEngine främst använt för Cryteks egna projekt och således licensierades CryEngine 2 endast till ett litet antal tredje partens spelutvecklare [26].

År 2009 lanserade Crytek sin uppdaterade spelmotor under namnet CryEngine 3. Avsikten var att förse spelutvecklare med samma avancerade grafikrenderingsmetoder och basfunktionaliteter som användes i Cryteks egna spel. Två år senare släppte Crytek också ut en gratisversion kallad "CryEngine Free SDK" för icke-kommersiellt bruk [27]. År 2013 döpte de om spelmotorn till "CryEngine" utan versionsnumret [28]. Spelmotorn höll detta namn fram till den femte versionen.

Crytek släppte ut CryEngine V år 2016 med en "betala vad du vill" -affärsmodell [29] som främst öppnade möjligheter för mindre studior med små budgetar. Inga ytterligare avgifter krävs för att använda spelmotorn, men Crytek säljer träningskurser med månadsavgifter för oerfarna användare. Spel skapade m.h.a. CryEngine V är huvudsakligen programmerade i C++ eller C# [30], men ett flödesschemasystem som liknar UnrealEngine 4:s Blueprints Visual Scripting är också tillgängligt för icke-programmerare [31].

4. Konstruktion av en jämförelsematris

Den generella strukturen av jämförelsen av de valda 3D-spelmotorerna utförs delvis enligt en modell som användes av Patrasitidecha [32] i en jämförelse av mobila spelmotorer. Målet är att konstruera en jämförelsematris som klart och tydligt lägger fram likheter och skillnader mellan motorerna. För att kunna göra detta måste jämförelsekriterierna först definieras. När kriterierna är definierade kan jämförelsen utföras, och efter det kan matrisen konstrueras.

4.1. Definiering av jämförelsekriterier

Enligt definitionen i kapitel 2.1. kan en spelmotor beskrivas som en samling sub-motorer. Detta medför att om de valda spelmotorerna ska jämföras måste deras sub-motorer jämföras med varandra. Stort sett har Unity, UnrealEngine och CryEngine dock liknande sub-motorer med liknande egenskaper. För att se skillnaderna måste alltså sub-motorernas inbyggda funktioner jämföras.

Eftersom de valda spelmotorerna kan användas av både nya och erfarna spelutvecklare måste också användarupplevelsen tas i beaktande. Tillgång till handledning och kodexempel är ovärderliga för oerfarna programmerare, medan veteraner i branschen värdesätter dokumentation och teknisk support. Även programmeringsspråket kan vara den avgörande faktorn i ett företags val av spelmotor. De olika faktorerna gällande användarupplevelsen ska alltså jämföras.

Utvecklingsplattformen spelar en stor roll i valet av en lämplig spelmotor. Om spelutvecklarna t.ex. använder Windows-datorer och en motor endast finns tillgängligt på macOS kommer de inte att investera i nya datorer utan söker istället efter andra, mer lämpliga spelmotorer. Exportplattformarna kan vara lika viktiga. Utvecklare och förlag som vill att spelet ska nå en så stor mängd konsumenter som möjligt väljer en spelmotor som stöder t.ex. både PC och spelkonsoler eller både iOS och Android. De nyanlända VR-plattformarna har dessutom väckt spelindustrins intresse och om spelmotorn stöder dem kan det också vara avgörande. Med andra ord måste både utvecklings- och exportplattformar jämföras.

Alla företag försöker gå på vinst. Det är alltså viktigt att en spelutvecklare väljer en spelmotor med ett pris som lämpar sig för slutproduktens förutspådda inkomster. Den sista jämförelsekategorin är spelmotorernas pris.

4.2. Jämförelse

Jämförelsen är indelad i de fyra ovannämnda kategorierna. Varje kategori tar upp de viktigaste delarna och jämför deras implementationer i de valda spelmotorerna. Egenskaper som ger identiska resultat i dem alla kan tas upp i jämförelsen, men inkluderas inte i jämförelsematrisen.

4.1.1. Jämförelse av inbyggda funktioner

I *Game Engine Architecture* [4] listar Gregory ett antal vanliga sub-motorer. Alla 3D-spelmotorer förväntas inkludera åtminstone en grafikrenderingsmotor, ett ljuduppspelningssystem, en fysikmotor samt nätverksfunktioner. Grundläggande funktioner

för artificiell intelligens anses vara ett plus. Dessa fem sub-motorer ska undersökas och deras funktioner i de valda spelmotorerna ska jämföras.

Unity 5, Unreal Engine 4 och CryEngine V använder DirectX- och OpenGL-gränssnitten för grafikrendering [33][34][35]. De tre motorerna har mycket liknande ljus-, skugg- och partikeleffekter, och kvaliteten på slutproduktens grafiska output beror huvudsakligen på 3D-modell- och texturartisterna. En intressant skillnad är att Unity 5 och Unreal Engine 4 kan även användas för tvådimensionella spel [36][37], medan CryEngine V endast lämpar sig för tredimensionella projekt.

Ljudmotorn är implementerad annorlunda i varje valda spelmotor. Unity 5 stöder flera olika filformat [38] medan Unreal Engine 4 har endast stöd för WAV-filer utan insticksprogram [39]. CryEngine V har däremot inget inbyggt stöd för ljuduppspelning, utan kräver "middleware" [40], ett typs program som kan liknas till en extern sub-motor skapad av en tredje part.

För att ett spel ska bete sig som förväntat behövs en realistisk fysiksimulation. Eftersom fysiken är universal kan alla utvecklare i teorin använda samma fysikmotor till sina ändamål. Det finns dock ej någon perfekt fysikmotor, så de olika spelmotorerna har implementerat sina egna lösningar. Både Unity 5 och Unreal Engine 4 använder Nvidias populära PhysX -motor för fysikberäkningar [41][42], medan Crytek har utvecklat CryPhysics, en egen fysikmotor för CryEngine V [43].

Nätverkskommunikation är en mycket viktig del av spelindustrin. Online skjut- och rollspel är väldigt populära och företagen bakom dem tjänar stora summor pengar på grund av det. Alla moderna spelmotorer förväntas inkludera funktioner för onlinespel. Unity 5 inkluderar ett ramverk för grundläggande nätverkskommunikation. Den bygger sig på en elementär server-klient struktur [44] som inte stöder etablerade onlinesystem utan extra integreringsarbete. Unreal Engine 4 har däremot en online-modul som stöder alla stora online-plattformar, t.ex. Steam, Xbox Live och PlayStation Network, förutsatt att utvecklaren har tillgång till den önskade plattformens SDK [45]. CryEngine V stöder endast Xbox Live, PlayStation Network och normal server-klient kommunikation [46].

Funktioner för artificiell intelligens inkluderas inte alltid i spelmotorer, men de tre valda spelmotorerna har sådana. Unity 5 har inbyggda pathfinding-algoritmer för navigering i 3D-rum [47], men mer komplicerade beteenden måste utvecklaren programmera själv. Unreal Engine 4 har flexibla beteendemönster kan editeras i ett visuellt trädsystem [48]. Trädet sköter om navigering samt andra händelser. CryEngine V har däremot förutom de tidigare nämnda funktionerna också färdigt programmerade fiendarketyper som lämpar sig till användning i skjutspel [49].

4.1.2. Jämförelse av användarupplevelse

I programvara är det viktigt att användarupplevelsen är god. Utvecklaren bör komma åt robust dokumentation och andra hjälpmedel utan hinder. I denna del jämförs dokumentationens kvalitet, handledningar, förekomsten av exempelprojekt för nybörjare, tillgängligheten av tekniska supporten samt programmeringsrelaterade frågor. Det är värt att nämna att alla valda spelmotorer har resursbutiker var utvecklare kan köpa grafik-, ljud och kodresurser, men eftersom de är mycket lika tas de inte med i jämförelsen.

Dokumentationen för en spelmotor ska vara detaljerad och lätt att använda. Dokumentationen för både Unity 5 och Unreal Engine 4 klarar av dessa mål med välstrukturerade navigeringsfält och välformulerade beskrivningar. Innehållen omfattar allt som en utvecklare behöver veta för att använda dessa spelmotorer. CryEngine V:s dokumentation är aningen förvirrande. Navigationsfältet är delat i tre delar: manualen för gamla versioner av CryEngine som innehåller en massa väldokumenterad information, manualen för CryEngine V som endast beskriver de nya funktionerna och till sist tekniska dokumentationen som innehåller både en guide för spelmotorns källkod och hjälp för programmering i CryEngine V. Det att hjälpen för att använda spelmotorn befinner sig under tekniska dokumentationen och inte under manualen för CryEngine V är inte intuitivt. Dessutom har de flesta artiklarna en massa länkar till sub-artiklar istället för text, så för att komma åt den eftersökta informationen måste man först navigera genom ett antal bristfälliga sidor.

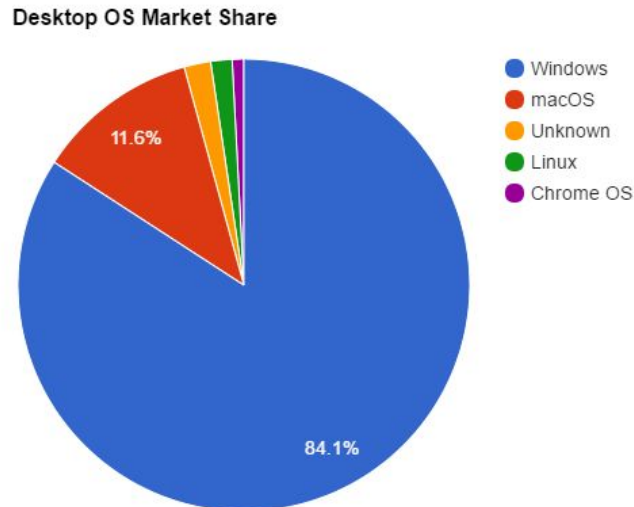
För en ny spelutvecklare är handledning och exempelprojekt bra sätt att komma igång. De erbjuder praktiska exempel på hur spelfunktioner implementeras med tydliga förklaringar. Handledningen för Unity 5 finns tillgängligt på internet i textform [50]. Ett exempelprojekt kan också installeras i samband med spelmotorn. För Unreal Engine 4 finns det handledning i både video- och textformat [51]. Dessutom kommer spelmotorn med färdiga mallar för spel av flera olika genrer för att utvecklaren kommer snabbare igång. Crytek har berett videohandledning för CryEngine V [52]. Spelmotorn har inga inbyggda exempelprojekt, men CryEngine-communityn har skapat en hel del sådana.

Angående teknisk support, Unity Technologies erbjuder på professionell support för en avgift [53]. Utvecklaren kan också söka hjälp på Unitys internetforum som naturligtvis är kostnadsfritt. Epic Games har en avgiftsfri support sida för Unreal Engine 4 [54] medan hjälp för CryEngine V endast fås genom communityn [55].

De valda spelmotorerna har alla en kodeditor för att utvecklarna ska kunna programmera sina spel. I Unity 5:s kodeditor kan programmeraren använda antingen C# eller JavaScript [11], I Unreal Engine 4 används C++ [56] och i CryEngine V duger både C++ och C# [30]. Utöver det har Unreal Engine 4 och CryEngine V sina programmerbara flödesscheman för icke-programmerare [21][31].

4.1.3. Jämförelse av utvecklings- och exportplattformar

Alla spelutvecklare använder datorer för att programmera sina spel. Enligt figur 1 har 84,1 % av alla persondatorer Windows som operativsystem. Det är alltså mycket viktigt att en spelmotor kan användas på Windows om motorns utvecklare vill nå så många spelutvecklare som möjligt. Det betyder ändå inte att den resterande andelen ska ignoreras. Spelutvecklare som använder Windows kan använda alla jämförda spelmotorerna, medan macOS- och Linux-användare endast kan köra Unity 5 och Unreal Engine 4 [57][34][58].



Figur 1: marknadsandelar för olika operativsystem. Källa för data: StatCounter [60].

Förutom utvecklingsplattformar är också exportplattformar viktiga. Många spelutvecklare vill att slutprodukten spelas av så många personer som möjligt oberoende av vilken plattform den kör på. Därför är stöd för många olika exportplattformar endast en bra sak. De olika plattformarna kan delas upp i kategorier. Den första kategorin i jämförelsen är datorer och webbplattformar. Den omfattar vanliga operativsystem för persondatorer samt teknologier som tillåter exekvering av spel i webbläsare. Den andra kategorin är spelkonsoler, både TV-konsoler och handhållna. I följande kategori jämförs stöd av mobila operativsystem. Den sista kategorin behandlar VR- och AR-glasögon som kan användas i samband med några plattformar.

Eftersom Unity 5 och Unreal Engine 4 kan användas på Windows, macOS och Linux är det inte överraskande att de också kan exportera spel till dessa operativsystem. Dessutom stöder de även HTML5-drivna webbläsare med hjälp av WebGL-programbiblioteket [2][59]. Trots att CryEngine V endast kör på Windows-datorer stöder exportmodulen förutom Windows också Linux, men den kan inte exportera spel till webbplattformar.

Spelkonsoler är populära p.g.a. att man kan spela de senaste spelen på dem utan att behöva investera i en dyr dator. De tre valda spelmotorerna stöder Xbox One och PlayStation 4 [2][59][35]. Unreal Engine 4 och Unity 5 lade nyligen till stöd även till Nintendo Switch.

Handhållna konsoler är fortfarande ganska populära fastän mobilappmarknaden blomstrar. Till skillnad från mobilappar är spelen som är skapade för handhållna konsoler utvecklade med avsikten att erbjuda konsol-liknande spel på en mindre prestandakrävande, bärbar maskin. Unity 5 är den enda spelmotorn av de tre valda som stöder handhållna konsoler, specifikt Nintendo 3DS och PlayStation Vita [2].

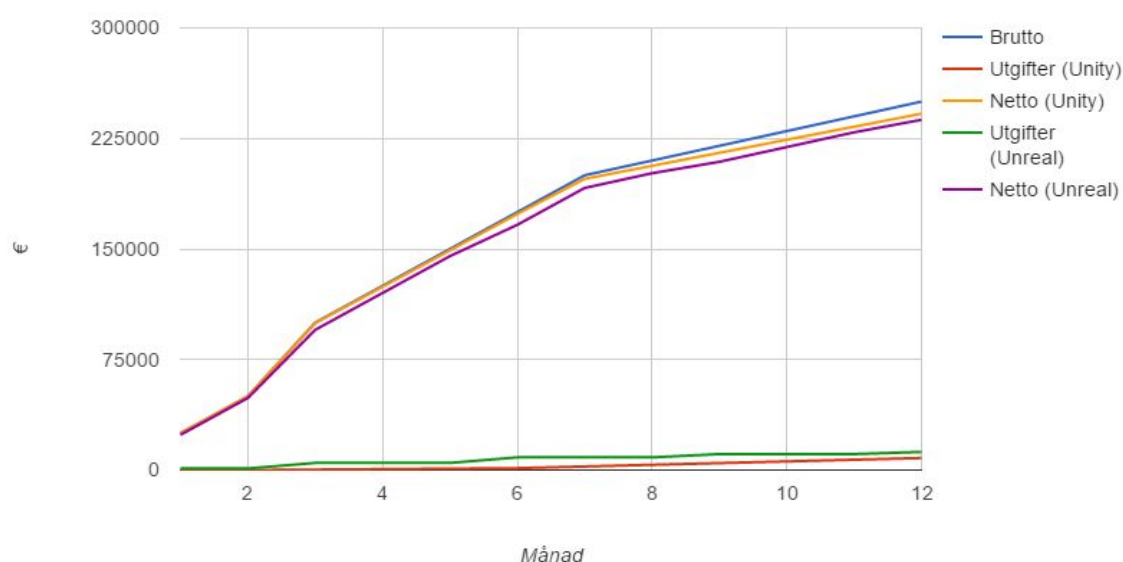
De mobila plattformarna har blivit allt populärare inom spelindustrin. De erbjuder små, snabba spel som är designade för att bli spelade lite i taget. Ofta innehåller de också en massa s.k. "mikrotransaktioner", alltså avgifter för att fortsätta spela eller för att ersätta skicklighet. CryEngine V var inte utvecklat för att stöda mobila plattformar [35], men de två andra stöder både Android och iOS [2][59]. Utöver det kan Unity 5 också exportera spel till de mindre populära operativsystemen Windows Phone, Tizen och Fire OS.

På grund av VR- och AR-glasögonens växande popularitet har Unity 5 och Unreal Engine 4 en konkurrensfördel jämfört med CryEngine V. CryEngine V stöder enbart Oculus Rift [35], medan Unity 5 och Unreal Engine 4 stöder alla kända dator-, konsol- och mobil-VR-plattformar [2][59]. Till dessa hör t.ex. SteamVR, PlayStation VR, Gear VR och Google VR. Dessutom är Unity 5 för tillfället den enda spelmotorn av de tre valda som stöder AR-plattformar, specifikt Microsofts HoloLens-glasögon.

4.1.4. Jämförelse av pris

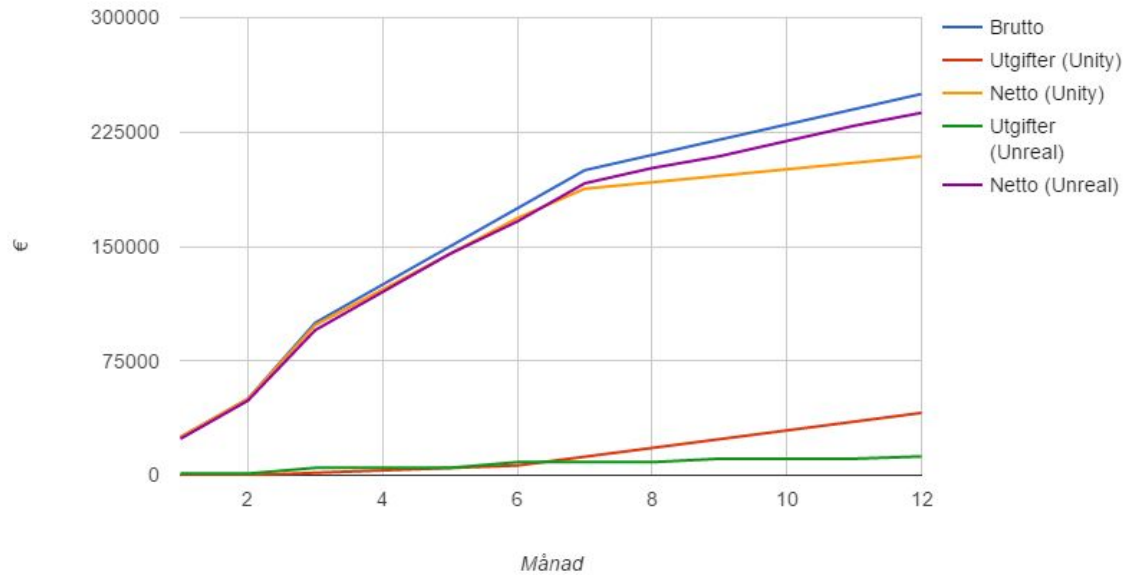
Alla tre spelmotorer har affärsmodeller som är speciellt lockande för nya spelutvecklare. Unity 5 är gratis tills en inkomstgräns på \$ 100 000 har nåtts, varefter den kostar 32 € per användare per månad. Om inkomsterna överskrider \$ 200 000 är månadsavgiften 115 € per person [12]. Unreal Engine 4 är gratis men 5 % av inkomsterna betalas var tredje månad åt Epic Games när en inkomst på \$ 3000 överskrider [20]. CryEngine V kostar så mycket som utvecklaren själv är villig att betala [29]. Den kan potentiellt vara den billigaste eller den dyraste spelmotorn och därmed har den alltför många okända variabler för att kunna bli rättvist jämförd med de andra spelmotorerna. På grund av detta är CryEngine inte ett intressant jämförelseobjekt.

För att jämföra Unity och Unreal Engine har skribenten skapat graferna i figur 2 och figur 3. Graferna representerar en hypotetisk scenario var spelutvecklarestudio med 10 respektive 50 programmerare har lanserat sina spel och förtjänar pengar på dem under en period på 12 månader. Grafen beaktar endast totala bruttoinkomsterna av spelen samt totala utgifterna och totala nettoinkomsterna som beror på om studiorna använde Unity 5 eller Unreal Engine 4 som spelmotor. Inkomstvärdena som används är påhittade för att tydligare demonstrera poängen och är inte baserade på något. Utgifterna kalkylerades enligt licensavgiftsvillkoren för vardera spelmotor.



Figur 2: graf över inkomster och utgifter med 10 programmerare

Med att avläsa grafen i figur 2 kan man se att Unreal Engine 4 är dyrare än Unity 5 om företaget har 10 programmerare. Det enda sammanhanget där Unity 5 kunde bli dyrare än Unreal Engine 4 var om inte ett enda exemplar av spelet såldes på flera månader, men p.g.a. att det är mycket osannolikt att sådant skulle hända i verkligheten inkluderades det scenariot inte i grafen. Hur skiljer sig detta resultat från det av ett företag med 50 programmerare?



Figur 3: graf över inkomster och utgifter med 50 programmerare

I grafen i figur 3 ser man att Unity 5 blir i långa loppet mycket dyrare än Unreal Engine 4 när flera och dyrare licenser krävs. Från dessa två grafer kan man dra slutsatsen att Unity 5 lämpar sig bättre åt studion med färre programmerare om priset är den avgörande faktorn.

4.3. Resultat

Med att samla ihop den relevanta informationen från jämförelsen ovan kan jämförelsematrisen konstrueras.

| | Unity 5 | Unreal Engine 4 | CryEngine V |
|---|--|--|--|
| Inbyggda funktioner | | | |
| Grafikrendering | 3D, 2D | 3D, 2D | 3D |
| Ljuduppspelning | .mp3, .ogg, .wav, .aiff, .mod, .it, .s3m, .xm | .wav, mer med plugins | Endast med middleware (fmod, Wwave, o.s.v.) |
| Fysikmotor | PhysX | PhysX | CryPhysics |
| Nätverkskommunikation | Server-klient | Server-klient, Steam, Xbox Live, PSN | Server-klient, Xbox Live, PSN |
| Artificiell intelligens | Pathfinding | Beteendemönster, pathfinding | Fiendarketyper, beteendemönster, pathfinding |
| Användarupplevelse | | | |
| Dokumentation | Omfattande | Omfattande | Bristfällig |
| Handledning | Text | Video och text | Video |
| Exempelprojekt | Ett inbyggt exempel | Inbyggda mallar för många olika genrer | Tillgängligt online |
| Teknisk support | Professionell support för en avgift | Avgiftsfri professionell | Community-baserad |
| Programmeringsspråk | C#, JavaScript | C++ | C++, C# |
| Programmeringsplattform | Kodeditor | Kodeditor, Blueprints Visual Scripting | Kodeditor, Flow Graph Editor |
| Utvecklings- och exportplattform | | | |
| Utvecklingsplattform | Windows, macOS, Linux | Windows, macOS, Linux | Windows |
| Export för datorer | Windows, macOS, Linux | Windows, macOS, Linux | Windows, Linux |
| Export för webbplattformar | HTML5 (WebGL) | HTML5 (WebGL) | - |
| Export för spelkonsoler | PlayStation 4, Xbox One, Nintendo Switch, PlayStation Vita, Nintendo 3DS | PlayStation 4, Xbox One, Nintendo Switch | PlayStation 4, Xbox One |
| Export för mobila plattformar | Android, iOS, Tizen, Windows Phone, Fire OS | Android, iOS | - |
| Stöd för VR-plattformar | Oculus Rift, SteamVR, PlayStation VR, Gear VR, Google VR | Oculus Rift, SteamVR, PlayStation VR, Gear VR, Google VR | Oculus Rift |
| Stöd för AR-plattformar | HoloLens | - | - |
| Pris | | | |
| Pris | Gratis, 35 € / mån eller 155 € / mån, beroende på företagets inkomst | 5 % av spelets inkomst efter att \$ 3000 har nåtts. | "Betala vad du vill"-modell |

Tabell 1: Jämförelsematrix

Från jämförelsematrisen i tabell 1 kan man se att alla tre spelmotorerna har sina starka och svaga sidor. Med Unity 5 kan spelutvecklare nå en mycket stor konsumentmängd eftersom den stöder en stor mängd olika plattformar, men avancerade funktioner för artificiell

intelligens och online-spel kräver mer jobb från programmerarnas sida. Unreal Engine 4 har ett imponerande antal stödda funktioner, men inbyggda ljudfilskompatibiliteten är mycket begränsad och AR-glasögon stöds inte alls för tillfället. CryEngine V:s affärsmodell är rättvis mot utvecklare, men spelmotorn stöder endast ett litet antal exportplattformar.

5. Avslutande diskussion

I slutändan beror valet av spelmotor enbart på spelutvecklaren. Kraven på spel av olika typer kan skilja sig oerhört mycket, och en lämplig spelmotor måste väljas på basen av det. Spelmotorns pris och användarupplevelse samt plattformens prestanda måste beaktas och evalueras efter egna behov. Unity 5 lämpar sig för mindre studior som vill nå många olika plattformar samt utvecklare av mobilspel. Unreal Engine 4 används av flera stora spelstudior för dator- och konsolspel men har också väckt intresse bland nybörjare. Användningen av spelmotorer i CryEngine-familjen har minskat drastiskt sedan Unity 5 och Unreal Engine 4 introducerades med sina prisvärda affärsmodeller. Det är i sig ingen överraskning eftersom den stöder så få plattformar och kräver tredje partens moduler för några grundläggande funktioner.

Eftersom antalet valda spelmotorer var mycket liten skulle det slutliga resultatet inte stämma fullständigt om fler motorer introducerades i matrisen. Matrisen är dock ett fungerande sätt för att lätt presentera rå data för enkel jämförelse och för att väcka läsarens intresse i den mer detaljerade texten.

I skribentens åsikt är Unity 5 och Unreal Engine 4 de bästa alternativen av de tre valda spelmotorerna. Deras respektive communities är aktiva och hjälp finns tillgängligt vid behov. Variationen av de stödda utvecklings- och exportplattformarna är också mycket bra. Det finns egentligen ingen orsak att explicit välja CryEngine V över dem om syftet är att utveckla ett spel.

Referenser

- [1] S. Zerbst, O. Düvel. *3D Game Engine Programming*. Course Technology PTR, 2004.
- [2] Unity Technologies. "Multiplatform" [Online].
Tillgängligt: <https://unity3d.com/unity/multiplatform> [Hämtat: 21 februari 2017]
- [3] Epic Games. "What is Unreal Engine 4" [Online]. Tillgängligt:
<https://www.unrealengine.com/what-is-unreal-engine-4> [Hämtat: 7 mars 2017]
- [4] J. Gregory. *Game Engine Architecture*. A K Peters/CRC Press, 2009.
- [5] Unreal Engine. (12 mars 2015). *Creating the Open World Kite Real-Time Demo in UE4 | GDC 2015 Event Coverage | Unreal Engine* [Online Video]. Tillgängligt:
<https://www.youtube.com/watch?v=clakekAHQx0> [Hämtat: 7 mars 2017]
- [6] K-I. Friese et al. "Using Game Engines for Visualization in Scientific Applications" i *New Frontiers for Entertainment Computing*. Springer, 2008. pp. 11-22.
- [7] D. Fritsch, M. Kada. (juni 2004). *Visualization Using Game Engines* [Online].
Tillgängligt: https://www.researchgate.net/publication/225075846_Visualisation_using_game_engines [Hämtat: 16 februari 2017]
- [8] R. Stanton. *A Brief History of Video Games: From Atari to Xbox One*. Hachette UK, 2015.
- [9] H. Lowood. "Game Engines and Game History" i *History of Games International Conference Proceedings* [Online]. Kinephanos, 2014. Tillgängligt:
<http://www.kinephanos.ca/2014/history-of-games-international-conference-proceedings/> [Hämtat: 15 februari 2017]
- [10] Unity Technologies. (7 juni 2010). "Unity Technologies Celebrates 5 Years of Unity" [Online]. Tillgängligt: <https://unity3d.com/http://unity3d.com/company/public-relations/news/unity-5year-press> [Hämtat: 20 februari 2017]
- [11] J. Brodtkin. (3 juni 2013). "How Unity3D Became a Game-Development Beast" [Online]. Tillgängligt:
<http://insights.dice.com/2013/06/03/how-unity3d-become-a-game-development-beast/> [Hämtat: 20 februari 2017]
- [12] Unity Technologies. "Unity Store" [Online].
Tillgängligt: <https://store.unity.com/?currency=EUR> [Hämtat: 21 februari 2017]
- [13] Epic Games. (2015). "Unreal Engine 1" [Online].
Tillgängligt: <http://www.moddb.com/engines/unreal-engine-1> [Hämtat: 23 februari 2017]
- [14] M. Thomsen. (23 februari 2010). "History of the Unreal Engine" [Online].
Tillgängligt: <http://www.ign.com/articles/2010/02/23/history-of-the-unreal-engine> [Hämtat: 23 februari 2017]
- [15] Epic Games. (2017). "Unreal Engine 3" [Online].
Tillgängligt: <http://www.moddb.com/engines/unreal-engine-3> [Hämtat: 24 februari 2017]
- [16] Epic Games. (29 januari 2013). "Epic Games Releases "Epic Citadel" For Android" [Online]. Tillgängligt:
<https://www.epicgames.com/news/epic-games-releases-epic-citadel-for-android/> [Hämtat: 28 februari 2017]

- [17] IGN Staff. (5 november 2009). "Epic Games Announces Unreal Development Kit, Powered by Unreal Engine 3" [Online]. Tillgängligt: <http://www.ign.com/articles/2009/11/05/epic-games-announces-unreal-development-kit-powered-by-unreal-engine-3> [Hämtat: 26 februari 2017]
- [18] A. Burnes. (7 januari 2012). "Epic Reveals Stunning Elemental Demo, & Tim Sweeney On Unreal Engine 4" [Online]. Tillgängligt: <http://www.geforce.com/whats-new/articles/stunning-videos-show-unreal-engine-4s-next-gen-gtx-680-powered-real-time-graphics> [Hämtat: 28 februari 2017]
- [19] GameTrailers. (7 juni 2012). *Unreal Engine 4 - GT.TV Exclusive Development Walkthrough* [Online Video]. Tillgängligt: https://www.youtube.com/watch?v=MOvfn1p92_8 [Hämtat: 28 februari 2017]
- [20] Epic Games. "Unreal Engine End User License Agreement" [Online]. Tillgängligt: <https://www.unrealengine.com/eula> [Hämtat: 8 mars 2017]
- [21] Epic Games. "Blueprints Visual Scripting" [Online]. Tillgängligt: <https://docs.unrealengine.com/latest/INT/Engine/Blueprints/> [Hämtat: 10 mars 2017]
- [22] Epic Games. "UnrealScript" [Online]. Tillgängligt: <https://docs.unrealengine.com/udk/Three/UnrealScriptHome.html> [Hämtat: 9 mars 2017]
- [23] Nvidia. "Games: X-Isle: Dinosaur Island" [Online]. Tillgängligt: <http://www.nvidia.com/object/dinoisle.html> [Hämtat: 10 mars 2017]
- [24] C. Hall. (11 juli 2013). "The Story of Crytek: From X-Isle Through Redemption" [Online]. Tillgängligt: <http://www.polygon.com/features/2013/7/11/4503782/crytek-x-isle-redemption> [Hämtat: 10 mars 2017]
- [25] A. Leather. "Crysis - Did you upgrade?" [Online]. Tillgängligt: <https://www.bit-tech.net/blog/2009/12/15/crysis-did-you-upgrade/> [Hämtat: 12 mars 2017]
- [26] Crytek. "CryENGINE® 2" [Online]. Tillgängligt: <http://www.crytek.com/cryengine/cryengine2/overview> [Hämtat: 12 mars 2017]
- [27] Crytek. "CryENGINE 3: Free SDK" [Online]. Tillgängligt: <http://www.moddb.com/engines/cryengine-3/downloads/cryengine-3-free-sdk> [Hämtat: 12 mars 2017]
- [28] T. Phillips. "Watch Crytek demo its next-gen CryEngine" [Online]. Tillgängligt: <http://www.eurogamer.net/articles/2013-08-22-watch-crytek-demo-its-next-gen-cryengine> [Hämtat: 12 mars 2017]
- [29] Crytek. "Crytek Unveils All-New CRYENGINE V and Community-Centered "Pay What You Want" Model" [Online]. Tillgängligt: <http://www.crytek.com/news/crytek-unveils-all-new-cryengine-v-and-community-centered-pay-what-you-want-model> [Hämtat: 14 mars 2017]
- [30] M. Endres, A. Lang. "CRYENGINE Code Tutorials" [Online]. Tillgängligt: <http://docs.cryengine.com/display/CEPROG/CRYENGINE+Code+Tutorials> [Hämtat: 14 mars 2017]
- [31] M. Endres, W. A. Haan. "Flow Graph Editor" [Online]. Tillgängligt: <http://docs.cryengine.com/display/CEMANUAL/Flow+Graph+Editor> [Hämtat: 14 mars 2017]
- [32] A. Patrasitidecha. *Comparison and evaluation of 3D mobile game engines*. Chalmers University of Technology, 2014.

- [33] Unity Technologies. "DirectX 11 and OpenGL Core" [Online]. Tillgängligt: <https://docs.unity3d.com/Manual/UsingDX11GL3Features.html> [Hämtat: 27 mars 2017]
- [34] Epic Games. "Hardware and Software Specifications" [Online]. Tillgängligt: [https://docs.unrealengine.com/latest/INT/GettingStarted/Recommended Specifications/](https://docs.unrealengine.com/latest/INT/GettingStarted/RecommendedSpecifications/) [Hämtat: 27 mars 2017]
- [35] Crytek. "Platforms" [Online]. Tillgängligt: <https://www.cryengine.com/features/platforms/> [Hämtat: 12 mars 2017]
- [36] Unity Technologies. "Unity 2D" [Online]. Tillgängligt: <https://docs.unity3d.com/Manual/Unity2D.html> [Hämtat: 27 mars 2017]
- [37] Epic Games. "Paper 2D" [Online]. Tillgängligt: <https://docs.unrealengine.com/latest/INT/Engine/Paper2D/index.html> [Hämtat: 27 mars 2017]
- [38] Unity Technologies. "Audio files" [Online]. Tillgängligt: <https://docs.unity3d.com/Manual/AudioFiles.html> [Hämtat: 27 mars 2017]
- [39] Epic Games. "Audio Files" [Online]. Tillgängligt: <https://docs.unrealengine.com/latest/INT/Engine/Audio/WAV/> [Hämtat: 27 mars 2017]
- [40] Crytek. "ATL - Audio Translation Layer" [Online]. Tillgängligt: <http://docs.cryengine.com/display/SDKDOC2/ATL+-+Audio+Translation+Layer> [Hämtat: 27 mars 2017]
- [41] Unity Technologies. "Physics in Unity 5.0" [Online]. Tillgängligt: <https://docs.unity3d.com/Manual/UpgradeGuide5-Physics.html> [Hämtat: 28 mars 2017]
- [42] Epic Games. "Physics Simulation" [Online]. Tillgängligt: <https://docs.unrealengine.com/latest/INT/Engine/Physics/index.html> [Hämtat: 28 mars 2017]
- [43] M. Pinard, A. Lang. "CryPhysics" [Online]. Tillgängligt: <http://docs.cryengine.com/display/CEPROG/CryPhysics> [Hämtat: 28 mars 2017]
- [44] Unity Technologies. "Setting up a Multiplayer Project from Scratch" [Online]. Tillgängligt: <https://docs.unity3d.com/Manual/UNetSetup.html> [Hämtat: 28 mars 2017]
- [45] Epic Games. "Online Subsystem Overview" [Online]. Tillgängligt: <https://docs.unrealengine.com/latest/INT/Programming/Online/index.html> [Hämtat: 31 mars 2017]
- [46] M. Pinard, A. Johnson. "CryLobby" [Online]. Tillgängligt: <http://docs.cryengine.com/display/SDKDOC4/CryLobby> [Hämtat: 31 mars 2017]
- [47] Unity Technologies. "Navigation and Pathfinding" [Online]. Tillgängligt: <https://docs.unity3d.com/Manual/Navigation.html> [Hämtat: 31 mars 2017]
- [48] Epic Games. "Behavior Trees" [Online]. Tillgängligt: <https://docs.unrealengine.com/latest/INT/Engine/AI/BehaviorTrees/index.html> [Hämtat: 31 mars 2017]
- [49] A. Chen. "AI" [Online]. Tillgängligt: <http://docs.cryengine.com/display/SDKDOC4/AI> [Hämtat: 31 mars 2017]
- [50] Unity Technologies. "Tutorials" [Online]. Tillgängligt: <https://unity3d.com/learn/tutorials> [Hämtat: 31 mars 2017]
- [51] Epic Games. "Samples and Tutorials" [Online]. Tillgängligt: <https://docs.unrealengine.com/latest/INT/Resources/index.html> [Hämtat: 31 mars 2017]
- [52] Crytek. "CryEngine V Video Training Series" [Online]. Tillgängligt: <https://www.cryengine.com/tutorials> [Hämtat: 1 april 2017]

- [53] Unity Technologies. "Premium Support" [Online]. Tillgängligt: <https://unity3d.com/learn/premium-support> [Hämtat: 1 april 2017]
- [54] Epic Games. "Unreal Engine 4 Support" [Online]. Tillgängligt: <https://www.unrealengine.com/support> [Hämtat: 1 april 2017]
- [55] Crytek. "CRYENGINE Answers" [Online]. Tillgängligt: <https://answers.cryengine.com/index.html> [Hämtat: 1 april 2017]
- [56] Epic Games. "Introduction to C++ Programming in UE4" [Online]. Tillgängligt: <https://docs.unrealengine.com/latest/INT/Programming/Introduction/index.html> [Hämtat: 2 april 2017]
- [57] Unity Technologies. "System Requirements for Unity 5.6" [Online]. Tillgängligt: <https://unity3d.com/unity/system-requirements> [Hämtat: 2 april 2017]
- [58] M. Endres, D. Perkins. "System Requirements" [Online]. Tillgängligt: <http://docs.cryengine.com/display/CEMANUAL/System+Requirements> [Hämtat: 2 april 2017]
- [59] Epic Games. "Frequently Asked Questions (FAQ)" [Online]. Tillgängligt: <https://www.unrealengine.com/faq> [Hämtat: 2 april 2017]
- [60] StatCounter. (Februari 2017). "Desktop Operating System Market Share Worldwide" [Online]. Tillgängligt: <http://gs.statcounter.com/os-market-share/desktop/worldwide/> [Hämtat: 26 mars 2017]