

Beteckning: _____



Institutionen för matematik, natur- och datavetenskap

Jämförelse av animationslager och trax-editor

Jonas Wallin
juni 2009

Examensarbete, 15 högskolepoäng, C
Datavetenskap

Kreativ programmering
Examinator: Ann-Sofie Östberg
Medbedömare: Bengt Östberg

Jämförelse av animationslager och trax-editor

av

Jonas Wallin

Institutionen för matematik, natur- och datavetenskap
Högskolan i Gävle

S-801 76 Gävle, Sweden

Email:

nkp06jwn@student.hig.se

suxor1@hotmail.com

Abstrakt

Redan under stenåldern fanns illustrationer föreställande djur med ben som pekade åt alla möjliga håll för att ge intrycket av rörelse. Metoden för animering har sen dess utvecklats radikalt och blivit väldigt stor del av vår vardag. Att animationer blivit mer lättillverkade och billigare att producera har utökat användningsområdet betydligt. Syftet med här arbetet är att ge ökad förståelse för hur man animerar i Maya. Det tar upp vad som gör en animering bra och vilka metoder animerare använder sig av när de framställer 3D animation. Arbetet är inriktat på framförallt två stycken olika verktyg i Maya som används vid animation, Trax-editorn och animationslager. En komparativ analys har gjorts av båda dessa verktyg. Animationslager är en väldigt ny funktion och det har därför inte gjorts någon liknande undersökning tidigare. Arbetet beskriver processen för två olika animationer som gjorts med hjälp av trax-editorn och animationslager. Slutsatsen av dessa test var att de ger animeraren möjligheten att snabbt ändra animationer väsentligt på ett väldigt smidigt sätt. Båda verktygen visade sig vara väldigt lika. Den största upptäckten var att animering går att utföra lika bra utan hjälp av animationslager och trax-editorn.

Nyckelord: Animering, Animationslager, Trax-editor, Rigging, Overview.

Innehåll

1 Inledning	1
1.1 Bakgrund	1
1.2 Syfte	1
1.3 Frågeställning	1
1.4 Avgränsning	1
2 Teoretisk Bakgrund	2
2.1 Animation	2
2.1.1 Squash and Stretch	2
2.1.2 Timing and Motion	2
2.1.3 Anticipation	3
2.1.4 Staging	3
2.1.5 Follow Through och Overlapping Action	3
2.1.6 Straight Ahead Action och Pose-to-Pose Action	3
2.1.7 Slow In and Out	4
2.1.8 Arcs	4
2.1.9 Exaggeration	4
2.1.10 Secondary Action	4
2.1.11 Appeal	4
2.1.12 Solid Drawing	5
2.2 Riggning	5
2.2.1 Handtag	6
2.2.2 Kontrollobjekt	6
2.3 Graf-editorn	7
2.4 Animationslager	8
2.4.1 Override-lager	8
2.4.2 Weight	8
2.5 Trax-editor	9
2.5.1 Character Set	10
2.5.2 Constraints	10
2.5.3 Relativ och absolut	11
2.5.4 Blend	12
3 Utförande	12
3.1 Animering 1	12
3.1.1 Animering 1 med trax-editorn	13
3.1.2 Animering 1 med animationslager	14
3.2 Animationstest 2	14
3.2.1 Animering 2 med trax-editorn	15
3.2.2 Animering 2 med animationslager	15
4 Resultat och diskussion	16
4.1 Resultat av animationstest	16
4.1.1 Animationstest 1	16
4.1.2 Animationstest 2	16
4.2 Diskussion	16
5 Slutsatser	17
5.1 Fortsatt arbete	18
Referenser	19

1 Inledning

1.1 Bakgrund

De första fynden människan gjort som är närbesläktade med animation är redan från stenåldern i form av grottmålningar. Dessa uråldriga illustrationer föreställde djur med ben som pekade åt alla möjliga håll för att ge intrycket av rörelse. Metoden för animering har sedan dess utvecklats radikalt och blivit en stor del av vår vardag, framförallt som underhållning genom film och spel, men även i reklam och vinjetter. Att animationer blivit mer lättillverkade och billigare att producera har utökat användningsområdet betydligt. 3D-visualisering och animering tillämpas nuförtiden mycket för presentationsmaterial i till exempel byggnadsprojekt.

1.2 Syfte

Syftet med här arbetet är att ge ökad förståelse för hur man animerar i Maya. Arbetet är inriktat på animering eftersom det finns ganska begränsad information om detta område, jämfört med till exempel modellering. Valet av just trax-editorn och animation-layer beror på att dessa verktyg är viktiga för att uppnå en hög nivå av skicklighet för animering i Maya. Animationslager är en ny funktion och det har därför inte gjorts någon liknande undersökning tidigare. Detta arbete bör ge en bra grund för hur man arbetar med dessa verktyg.

1.3 Frågeställning

- Vilka är de vanligaste metoderna för att framställa en animation?
- Vilka verktyg behöver en animerare för att kunna animera?
- Vad har trax-editorn att erbjuda en animerare?
- Vad har animationslager att erbjuda en animerare?

1.4 Avgränsning

Jag har valt att ta upp de mest vitala delarna av animationsprocessen, såsom riggning, verktyg, animationsprinciper och animationsmetoder. Arbetet kommer inte att ta upp någonting av 2D animation utan inriktas endast på 3D animation. Området som har störst fokus är verktygen grafeditorn, trax-editorn och animationslager.

2 Teoretisk Bakgrund

2.1 Animation

En animation är en mängd bilder som snabbt visas i följd efter varandra. Genom att ändra positioner på till exempel objekt i bilderna ger det åskådaren illusionen av att det finns rörelse i animationen [1]. Illusionen blir bäst när animationen uppdaterar bilderna mer än 24 gånger per sekund. Detta beror på att det är den uppdateringshastigheten det mänskliga ögat har. Har en animation en uppdateringshastighet som är färre än 24fps (Frames per second) så kommer den att uppfattas som ryckig och inte lika tillfredställande att se på.

Det finns 12 stycken olika principer för traditionell animering. Dessa formades på 1930 talet av Walt Disney studios. Syftet med dem var att göra animering, framförallt karaktärsanimering mer realistisk och underhållande. [4] [3]

2.1.1 Squash and Stretch

För en animerare är det viktigt att veta att formen för organiska objekt i viss grad deformeras. Ta dina bicepsmuskler till exempel, när du drar ihop armarna så spänns bicepsmuskulerna och står ut. När du istället sträcker ut armarna så förlängs bicepsmuskulerna och står längre inte ut. En viktig regel är att volymen för objektet ska vara konstant, annars kan objektet uppfattas som det förminskas när det mosas och växer när det stretchas. Principen är bra att tänka på vid animering av snabba rörelser av objekt. Är rörelsen långsam så överlappar objektet sig själv mellan bilderna och ögat jämnar och mjukar ut rörelsen. Men är rörelsen så pass snabb att objektet inte överlappar mellan bilderna kommer ögat att separera bilderna och inte uppfatta animationen som en rörelse. Lösningen kan därför vara att stretcha objektet så att det överlappar, vilket ger en mjuk animation.

2.1.2 Timing and Motion

Timing och hastigheten är vad som ger en rörelse dess syfte. Animeraren behöver låta handlingen och reaktionen i animationen ta lämplig mängd tid. Annars finns risken att åskådaren tappar intresse om det går för långsamt, eller att den inte förstår eller upptäcker händelsen ifall det går för fort.

Timing kan också påverka intrycket åskådaren får av ett objekt. Till exempel ger långsammare acceleration ett tecken på att objektet är tyngre.

Ett objekts rörelse kan också ge intrycket av tyngd. Föreställ dig en boll som studsar mot en kub. Detta ger intrycket att bollen är mycket lättare än kuben. Ifall bollen istället för att studsas skulle flytta på kuben skulle detta ge intrycket av att bollen är mycket tyngre än kuben.

2.1.3 Anticipation

Anticipation kan vara den anatomiska förberedelsen för en handling, till exempel att dra tillbaka foten för att ladda inför ett skott. Det är också bra för att förbereda åskådaren att något ska hända så att den har ögonen på rätt område av bilden. Exempel på detta är lyfta armarna och stirra på något innan de plockar upp det. Ett annat exempel är att stirra på någonting som inte är i bild och reagera på det innan handlingen visas för åskådarna. På så sätt kan animeraren lägga bildens fokus både på handlingen och reaktionen.

Anticipation kan även användas för att ge illusionen av tyngd och massa. En tung person kanske tar händerna på armstöden på en stol innan han reser sig medan en lättare person mer troligt bara skulle resa sig upp direkt.

2.1.4 Staging

Staging betyder iscensätta och det innebär att presentera en idé så att det är tydligt. En idé kan vara en händelse, en personlighet, ett uttryck eller ett humör. En viktig del av staging är att leda åskådarens öga till rätt det av bilden så att den inte missar någon händelse. Detta innebär att bara en idé kan visas åt gången, annars kanske åskådaren kollar på fel sak. Ett bra knep för att locka blicken på rätt plats är att ha rörelse där, ögat drar sig till rörelse i en annars stilla scen. I en scen där det mesta har rörelse så kommer ögat att dra sig till ett stilla objekt.

2.1.5 Follow Through och Overlapping Action

Follow through är den avslutande delen av en händelse. Ett bra exempel är när man kastar en boll – handen fortsätter rörelsen även efter att den släppt bollen. Rörelserna i ett komplext objekt som har flera delar som rör sig vid olika tillfällen och olika hastigheter. Vid en gångcykel till exempel så leder höfterna, följt av benen och sedan fötterna.

Overlapping action betyder att starta ytterligare en händelse innan den första har avslutats. Detta bibehåller intresset hos åskådaren eftersom det inte är någon döttid mellan händelserna.

2.1.6 Straight Ahead Action och Pose-to-Pose Action

Straight Ahead Action är när animatören animerar händelserna från början till slut utan att planera händelserna däremellan. Detta skapar mer spontana och fåniga animationer och används för vild och fartfylld action.

Pose-to-Pose Action är när animatören noggrant planerar ut animationen i förväg genom att till exempel väldigt grovt blocka ut sektionen i en händelse i poser, först den ursprungliga, några däremellan, och sedan den slutgiltiga posen. Sedan animerar animatören alla bilderna mellan dessa blocks så att det blir en fullständig animation. Detta används när scenen kräver mer tanke och när poserna och timingen är av mer betydelse.

2.1.7 Slow In and Out

Med detta refererar man till bilderna kring rörelsens maximum positioner. En konstant hastighet på ett objekt i rörelse kan ofta uppfattas som tråkigare och ibland mindre realistiskt än när hastigheten varierar vid extrem värdena. Till exempel en boll som studsar har högre hastighet precis innan den studsar och när den lämnar marken än när det har sitt maxvärde i luften.

I Maya uppfyller man detta ganska enkelt genom att påverka graferna i GraphEditor som beskriver animationens rörelse och hastighet. I exemplet med bollen så skulle grafen vara avrundat för y-axeln vid bollens maxvärde i luften.

2.1.8 Arcs

Alla handlingar med ytterst få undantag följer en relativt cirkulär rörelse. Detta gäller framförallt människor och djur. Detta beror på att rörelsen människor och djur gör utgår från lederna. Kroppsdelen som sitter i änden av en led kommer därför att följa ett cirkulärt spår utifrån den leden. Animationer som har arc-rörelser är i regel mer verklighetstroga och ser mycket bättre ut.

2.1.9 Exaggeration

Att överdriva vissa animationer kan ofta vara en väldigt bra idé när man animerar. Det kan få en komisk effekt, är härligare att se på och det förtydligar också animationen. Typiska handlingar att överdriva är ansiktsuttryck, gångstil, klumpighet etc.

Det är viktigt att vara försiktig när man överdriver en animation. Om man bara överdriver en sak finns risken att det står ut för mycket ur resten av animationen. Om man överdriver allting i scenen så kanske den uppfattas som alldeles för orealistisk och dramatiserad.

2.1.10 Secondary Action

En bra metod för att förstärka syftet med en animation är att applicera det i flera dimensioner. Om känslan animeraren vill inge är aggressivitet hos en karaktär så är det ofta väldigt bra att applicera detta på mer än bara ansiktet som exempel. Det kan även vara positivt att ge karaktären en aggressiv gångstil genom att till exempel göra den stampande, kraftfull och framåtlutande.

2.1.11 Appeal

Appeal betyder att det är något med animationen som åskådaren gillar att se. Det är motsvarigheten till en riktig skådespelares karisma. Utseendet på karaktären är huvudfaktorn till en animations appeal. Personligheten i karaktären ska återges i dess utseende, en snäll karaktär bör antagligen se söt och gullig ut, en skurk bör ha något som kännetecknar en sådan, till exempel en cigarr, ovärdade eller saknade tänder, mörka ögon etc.

Bra appeal fångar åskådarens intresse och behåller det.

2.1.12 Solid Drawing

Detta berör i princip bara 2D-animationer. Det är tekniken att ge en 2D-bild en 3D-känsla genom att skapa djup i bilden. Detta uppnås till exempel genom att rita ut skuggor.

Det som ska appliceras i alla dessa 12 principer är personlighet som är lockande för åskådaren. Det är nödvändigt att personligheten som man vill framhäva är konsistent i alla de olika principerna så att inga tveksamheter uppstår.

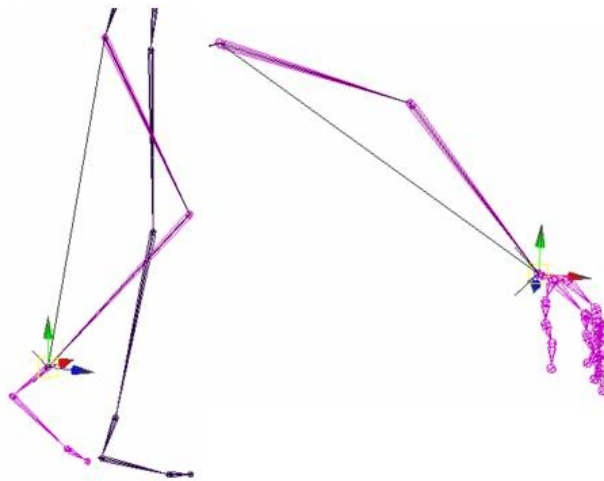
Det finns flera olika typer av animationer. Några exempel är stopmotion, frame-by-frame och datoranimation. Stopmotion är animation som skapas i huvudsak genom att man tar en bild på något som finns i verkligheten, sedan ändrar man på det och tar en ny bild. Detta upprepas tills man har en mängd bilder som kan sättas i en bildsekvens och på så sätt skapas en animation. Frame-by-frame bygger på att man förändrar en bild genom att till exempel rita om den. Detta går att göra för hand eller med hjälp av verktyg i datorn. Det finns både 2D- och 3D-frame-by-frame-animationer. Datoranimation är konsten att skapa animationer med hjälp av digitala medier i datorn. Datoranimation är den metoden som är absolut vanligast i dagsläge och även den som jag tänker beskriva ingående i detta arbete.

2.2 Riggning

3D-datoranimationer är mycket mer avancerade än de andra metoderna. Objekten som finns i 3D animationerna är modellerade i 3D-program. Autodesk: Maya är ett exempel på 3D-program som kan framställa 3D-miljöer. 3D-modellerade karaktärer tar väldigt lång tid att göra och därför är det inte effektivt att skapa om modellerna i annorlunda positioner, storlekar och gester. Detta är möjligt för 2D-miljöer och karaktärer som tar betydligt kortare tid att skapa, eftersom de har en dimension mindre att arbeta mot. För att kunna förändra en 3D-karaktär ger man den ett skelett. Detta skelett är osynligt men bundet till den synbara ytan av 3D-karaktären. Det är viktigt att man placerar skelettet på rätt ställe i förhållande till ytan så att lederna vrider karaktärens lemmar på ett realistiskt sätt. Att tillföra en karaktär ett skelett är en viktig del av pipeline för en 3D-animation med karaktärer, tillvägagångssättet kallar man för riggning.

2.2.1 Handtag

Skelettet är bara en del av processen mot en fullständig riggning. Efter att skelettet är korrekt placerad och ytan av karaktären bunden så är nästa steg att koppla så kallade handtag (Handles). Handtagen man kopplar till skelettet förenklar väldigt mycket för animeraren eftersom de ger möjligheten att kunna flytta flera leder i skelettet samtidigt på ett realistiskt sätt, förutsatt att handtagen har dragits rätt. Det är lämpligt att dra ett handtag till exempel för armarna, i det fallet från axeln till handleden (*Figur 1*). Detta kommer att göra rörelsen för den simulerade armen mer realistisk i sina rörelser då den kommer att vrida sin armbåge i förhållande till handledens position. Det finns flera stycken olika typer av handtag: IK handle och FK handle (Inverse Kinematics och Forward Kinematics) och spline handle för att nämna några. [2]



Figur 1. Inverse Kinematics applicerad på en arm och ett ben

2.2.2 Kontrollobjekt

Det är genom att påverka dessa handtag som animeraren kan manövrera karaktären.

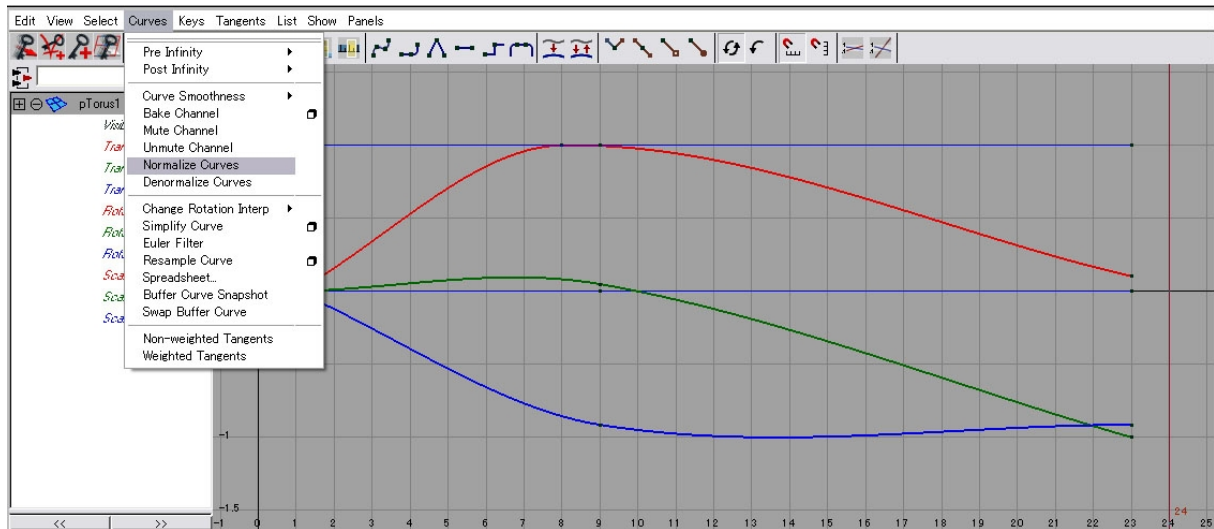
Det kan dock vara besvärligt att markera handtagen som ofta gömmer sig bland skelettets leder eller karaktärens geometri. Det går att förenkla denna process genom att koppla kontrollobjekt som består utav kurvor till handtagen. Kurvor visas inte vid den slutgiltiga framställningen av animationen (renderat material) och passar därför bra som byggmaterial för kontrollobjekten. Kontrollobjekten ska av praktiska skäl ha så funktionell form och placering att det är lätt att se och komma åt dem. Deras form kan även påminna om den del av karaktären som den representerar. Denna metod innebär att animeraren kan markera ett kontrollobjekt och genom att påverka den så påverkas även handtagen som är kopplade till kontrollobjektet. Denna teknik gör jobbet betydligt enklare för animeraren.

Det är vanligt att riggaren ställer in en funktion i ett kontrollobjekt som gör det möjligt för animeraren att påverka till exempel förflyttelse eller rotation av en eller flera leder genom att dra i en slider med värden. Detta kan vara mycket lättare än att förflytta dem manuellt i 3D-vyn.

2.3 Graf-editorn

Det finns ett verktyg i Maya som jag själv anser vara helt oombärligt när man animerar.

Verktyget heter Graf-editorn (*figur 2*) och beskriver alla animationer man skapat i kurvor.



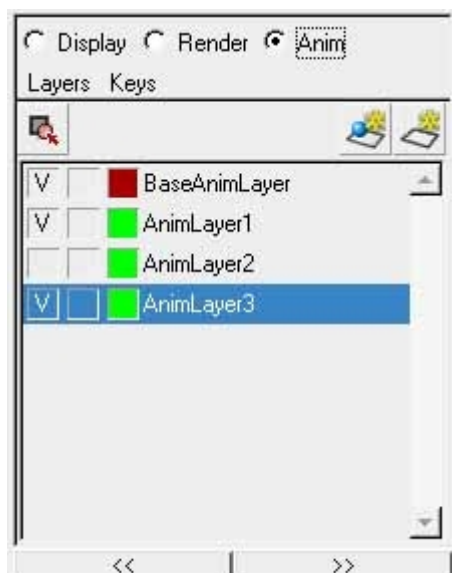
Figur 2. Bild på Graf-editorn i Maya. Kurvorna beskriver animationen.

Att framställa animationen i kurvor har väldigt stora fördelar. Alla keys man placerat ut i sin animation visas i grafeditorn på ett tydligt sätt. Det går enkelt att läsa av vid vilken frame de sitter och vilket värde de har. Det går att markera dessa keys och flytta på dem och på så sätt förändra animationen. Graf-editorn ger betydligt mer överblick av animationen än jämfört med att bara se vart det finns keys i timeslidern. Det är betydligt enklare att upptäcka brister i animationen. En metod för felsökning av vissa animeringsbrister är att leta i kurvorna efter oregelbundenhet. Ifall det är tänkt att en rörelse skall vara konstant så ska dess kurva vara rak till exempel. Ett föremål som kastas upp i luften har inte en tvär vändning när det faller tillbaka. Dess kurva skall därför vara mjuk och inte kantig.

Graf-editorn visar alla utplacerade keys i en animation för markerat kontrollobjekt. Detta gör det möjligt att markera flera keys och ändra dem genom att till exempel skala och flytta. Detta är ett smidigt sätt ifall man önskar att hela animationen skall påbörjas vid en annan frame eller ifall den skall vara snabbare eller långsammare.

2.4 Animationslager

Animationslager i Maya är ett mycket kraftfullt verktyg (figur3). Det gör tekniken att påverka, finslipa och ändra färdiga animationer mycket enklare. Den ursprungliga animationen som man vill påverka genom animationslager sparas i ett enskilt lager som heter BaseAnimation som standard. Den blir på så sätt s.k. non-destruktiv, vilket innebär att även om animeraren påverkar animationen genom att till exempel lägga till lager så finns den ursprungliga animationen kvar oförstörd i BaseAnimation lagret.



Figur3. Lager med animationer i animationslagerfönstret.

Det är väldigt enkelt att använda sig av animationslager. Ifall du har en animation med till exempel en karaktär som springer och du i efterhand vill göra så att han efter kanske en sekund vrider på huvudet och kollar in i kameran, så är det enda du behöver göra att markera kontrollobjektet för huvudet och tilldela det ett lager, sedan animerar du att han vrider huvudet mot kameran en sekund in i animationen. Så länge det här lagret är aktivt så kommer karaktären att vrida huvudet vid utsatt tid. Önskas huvudvridningen inte längre är det bara att släcka ned det lagret. Det går väldigt bra att exportera animationslager till en annan karaktär med liknande karaktärsrig. På så sätt behöver man inte animera om samma rörelse flera gånger.

2.4.1 Override-lager

Det är möjligt att göra ett s.k. override lager. Det innebär att när det lagret ligger överst i hierarkin så stryps alla andra inkommande kontakter från andra lager som innehåller som påverkar samma kontrollobjekt. Detta är väldigt hädigt när man vill göra en stor ändring i en animation under utvalda frames utan att påverka de som är innan och efter.

2.4.2 Weight

En annan viktig funktion i animationslager som används frekvent är den s.k. weight-funktionen. Weight-funktionen gör det möjligt för användaren att ändra hur mycket styrka animationerna i ett animationslager har på kontrollobjekten. Denna kontroll ges genom en slider under lagret. För att förklara hur detta fungerar kan jag visa med ett exempel. Du har ett animationslager som gör att en karaktär hoppar högre än vad den gör i den ursprungliga animationen. Har du styrkan på slidern i toppen så hoppar karaktären så mycket högre som animeringen i det

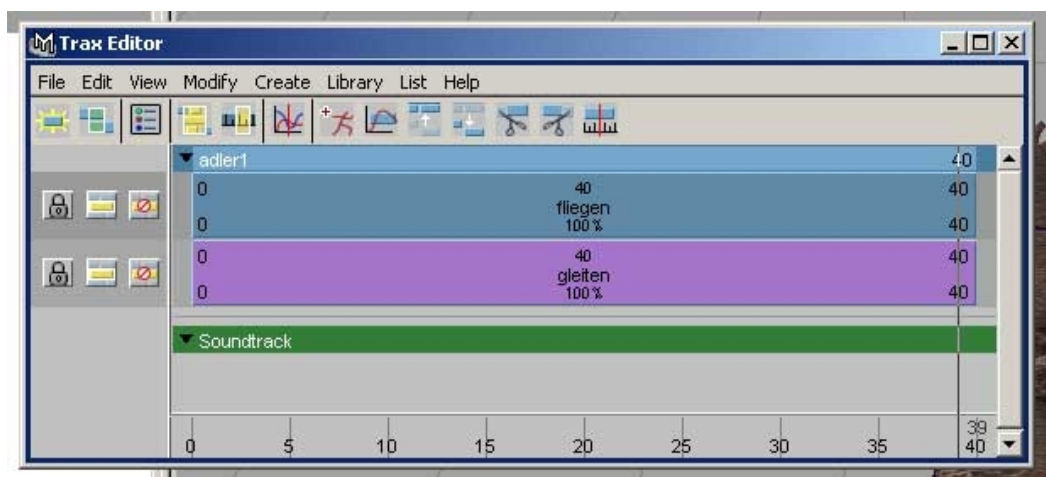
animationslagret beskriver. Har du styrkan istället vid hälften av maximalvärdet så kommer karaktären bara hoppa hälften så mycket. Detta är ett smidigt sätt att ändra sina animationer utan att nödvändigtvis behöva ändra några keyframes.

Det går att spara kodade uttryck i animationslager. Till exempel ett uttryck som gör att karaktären vibrerar på utvalda kontrollobjekt går att spara i ett lager som går att aktivera på önskade frames.

När man är nöjd över sina animationer som man sparar i olika lager går det att sammanställa dem till ett enda lager som man sedan kan spara som ett animationsklipp i trax-editorn. Hur trax-editorn fungerar förklaras i följande avsnitt.

2.5 Trax-editor

Trax-editorn är ett verktyg i Maya precis som animationslager. Det är ett eget fönster med flera spår i vilka man kan lägga sina animationsklipp (figur4). Animationsklippen lägger sig i spåren som rektangulära rutor, som är lika långa som animationen är lång. Genom att flytta på klippen så ändrar man startpunkten för animationen. Ifall man önskar att spela animationen i klippet långsammare eller snabbare så är det enda man behöver göra att skala det. Det går även att kopiera klippen och spara dem så att man kan infoga dem när de behövs. Det är lämpligt att spara animationer som går att upprepa i trax-editorn. Ett exempel på cykelbar animation är en gångcykel. I en korrekt gångcykel så är början av cykeln vid samma fas av steget som när det slutar. På detta vis går det att spela om samma animation hur många gånger man vill utan att det rycker till när animationen upprepas. Klippen som man sparar läggs i bibliotek så att man kan infoga dem när man önskar.



Figur 4. Animationerna lägger sig som klipp i trax-editorn.

Klippen som man lägger i trax-editorn behöver inte vara hela animationen som ska visas utan det kan vara precis som vid användandet av animationslager bara utvalda kontrollobjekts rörelser. På så sätt går det att styra över animationen väldigt mycket och smidigt.

2.5.1 Character Set

Det går att placera flera olika kontrollobjekt i en gemensam samling som kallas Character Set. Alla kontrollobjekt som tillhör en karaktär lägger man oftast in i ett Character Set så att man lätt kan hitta de animationsklipp som tillhör den. Det går att aktivera Character Set i trax-editorn. Aktiverar man ett Character Set så visas alla aktiva animationsklipp som tillhör det. Varje Character Set har ett eget bibliotek från vilket man kan hämta dess animationsklipp.

En gångcykel är ett typiskt animationsklipp som tillhör ett Character Set eftersom den berör flera kontrollobjekt.

2.5.1.1 Sub Character Set

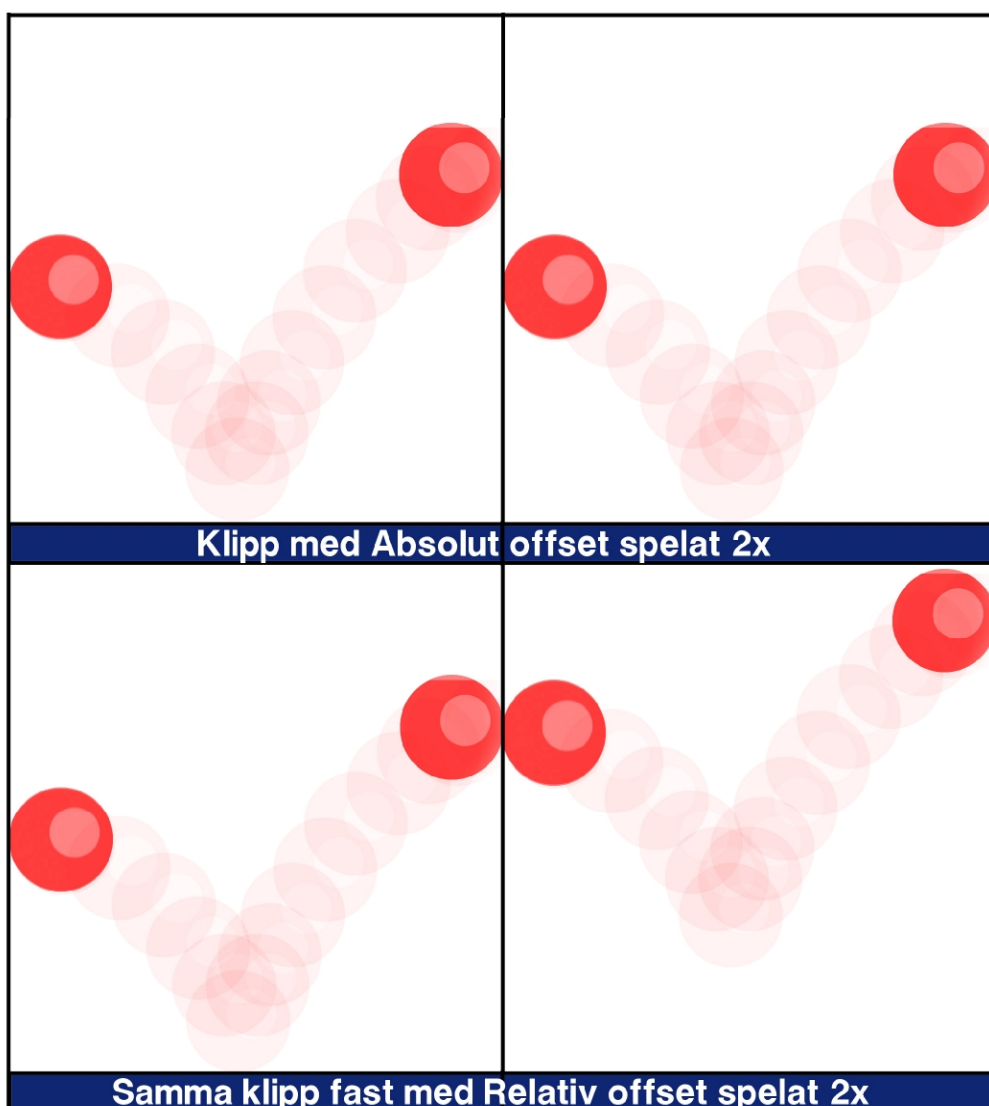
Det går att dela upp ett Character Set till flera s.k. Sub Character Set. Det innebär att en del av kontrollobjekten hör till ett Sub Character Set och de andra ett annat Sub Character Set. Animationerna för alla kontrollobjekt finns kvar i animationsklippen som tillhör det ursprungliga Character Set. Det som är effektivt med det här är att man kan påverka ett Sub Character Set utan att det andra ändras. På så sätt kan man behålla gångcykeln för till exempel underkroppen men ändra överkroppen.

2.5.2 Constraints

I Maya går det att göra kopplingar mellan objekt som kallas constraints [7]. En typ av constraint är aim-constraint. Adderar man en sådan mellan två objekt så kommer objektet rotera beroende på vart det andra objektet befinner sig. Detta är effektivt att utnyttja till exempel när en karaktär följer något med blicken. Constraints går också att spara i klipp precis som animationer. Det finns ett problem med en animation med en karaktär som följer något med blicken med hjälp av en aim-constraint. Det kan verka orealistiskt om någon följer något som rör på sig helt punktligt. Det kan ge intrycket att karaktären till och med styr föremålet med sin blick, eller att den vet vart föremålet ska ta vägen. Detta går att lösa genom att spara sitt aim-constraint i ett klipp. Gör man det går det att flytta constraint-klippet så att det ligger lite efter klippet med rörelsen för föremålet. Detta gör att det blir en fördröjning av karaktärens blick, vilket gör det mycket mer realistiskt.

2.5.3 Relativ och absolut

I klippen i trax-editorn går det att ändra om offsets för varje objekt som är påverkade ska vara relativ eller absolut. Skillnaden mellan dessa två är att absolut offset gör så att rörelserna i klippet uttrycker sig precis som innan de sattes i klipp, när det kommer till position och rotation. Ställer man in offset på relativ så kommer rörelsen i klippet baseras på positionen och rotationen kontrollobjektet har precis innan klippet. Vid relativ offset adderas alltså rörelsen i klippet på objektet istället för att börja om. Exemplet nedanför (*figur 5*) förtydligar detta.



Figur 5. Samma klipp som spelas två gånger efter varandra med olika offsets valda.

2.5.4 Blend

Ifall det är två klipp i trax-editorn som ligger efter varandra och det första klippet inte slutar som det andra klippet börjar så kommer animationen inte vara flytande och karaktärerna kommer att hoppa till en annan position när det andra klippet startar. Det finns en enkel lösning på detta som kallas blend. Blend är en funktion i trax-editorn som lägger sig mellan två klipp. Det är som ett slags autoregenererat klipp som interpolerar två klipp. Precis som ett vanligt klipp går det att ändra hur lång tid interpoleringen kommer att ta.

3 Utförande

Jag har gjort flera olika animationer som en metod för att jämföra skillnader mellan trax-editorn och animationslager.

3.1 Animering 1

Den första animationen (*figur 6*) jag gjorde består av en hjort, som är en gratisrigg från Internet[5]. Hjorten skall gå under hela animationen. Vid en viss tidpunkt skall den vrida huvudet och titta in i kameran och sedan vrida tillbaka huvudet igen. Denna huvudvridning skall göras i två versioner, en med hjälp av trax-editorn och en i animationslager

Jag började med att animera gångcykeln till hjorten.



Figur 6. Hjorten i den första animeringen som har tilldelats en gångcykel.

3.1.1 Animering 1 med trax-editorn

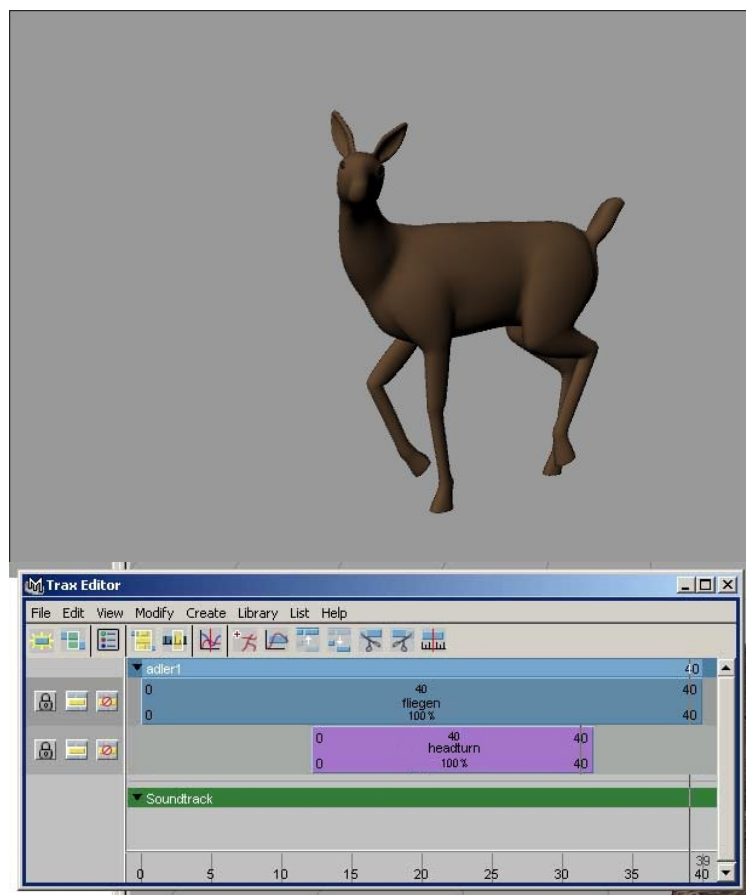
Jag började med att göra huvudvridningen med trax-editorn.

Det första jag gjorde var att spara ned gångcykeln i ett animationsklipp för hjortens Character Set. Med gångcykeln i ett eget animationsklipp kan jag enkelt lägga till rörelser i animationen under tiden hjorten går utan att hindra flödet i gångcykeln.

Jag animerade sedan huvudvridningen (*figur 7*) genom att sätta en key med kontrollobjekten för halsen och huvudet utan att flytta dem något. Denna keyframe markerar alltså startpunkten för vridningen. Sedan vred jag huvudet och halsen i position så att hjorten tittar in i kameran och satte en ny key. Jag satte ytterligare en key några keyframes fram med hjortens huvud fortfarande mot kameran. Denna key markerar hur länge hjorten skall titta mot kameran.

Nästa key jag satte var för när hjorten vänder tillbaka huvudet.

Det är viktigt att man inte animerar huvudvridningen när hjortens Character Set är aktivt, för då kommer alla aktiva animationsklipp som tillhör detta Character Set att strula.



Figur 7. Hjorten vridder huvudet mot kameran. Vridningen är definierad i ett animationsklipp.

Efter huvudvridningen var animerad och klar sparade jag ned animationsklippet i trax-editorn under hjortens Character Set. Med huvudvridningen i ett animationsklipp kan jag nu effektivt styra över huvudvridningen; när den skall börja, hur snabb den skall vara, hur länge han skall kolla mot kameran, etc.

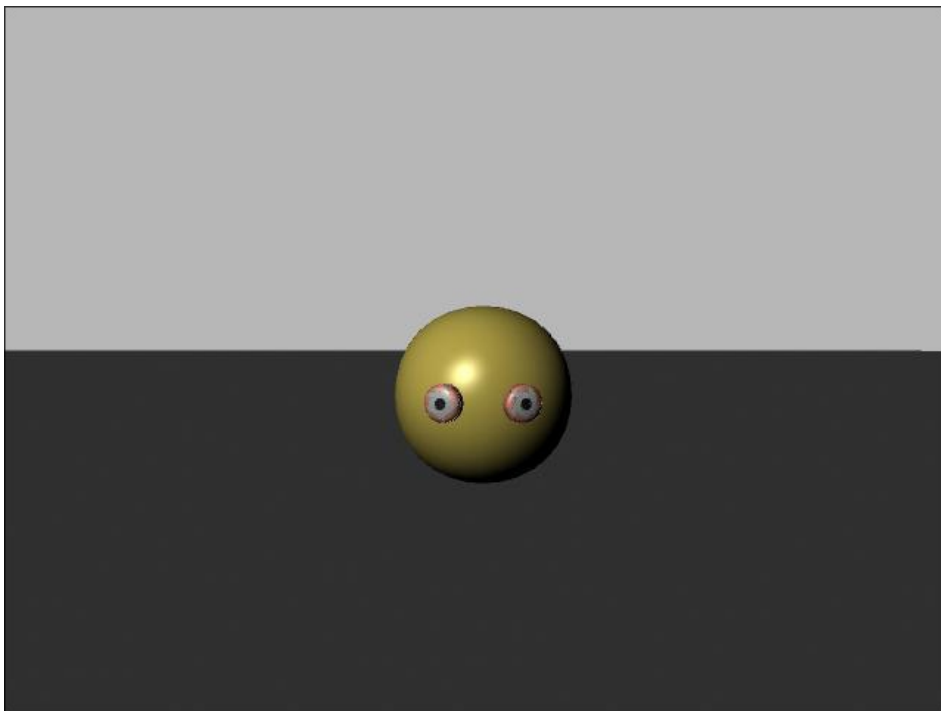
3.1.2 Animering 1 med animationslager

Den andra versionen av denna animation gjorde jag med hjälp av animationslager. Jag började med att markera alla hjortens kontrollobjekt och sparade ned dem i ett lager. På så sätt fick jag med hela gångecykeln i ett enda lager. Sedan markerade jag huvudet och halsen och sparade ned dem i ett annat lager som jag döpte *headTurn*. Med *headTurn* lagret aktivt animerade jag så att hjorten vrider huvudet mot kameran. Metoden för det är inte likadan när man arbetar med animationslager. Det första jag gjorde i version två var att sätta en *Zero Key Layer* vid den frame där jag vill att en ny rörelse, huvudvridning i det här fallet, skall börja. Några frames längre fram roterar jag huvudet och nacken i position och sätter en ny key. Jag låter huvudet och nacken vara i den positionen och sätter en ny key så många frames fram som jag vill att hjorten ska ha huvudet mot kameran. Det enda jag behöver göra för att hjorten skall vrida huvudet tillbaka till ursprungspositionen är att trycka på *Zero Key Layer* vid den frame jag vill att han ska ha huvudet tillbaka vridit. *Zero key Layer* tar, vid utvald frame, bort alla inkommande influenser från lagret, därför behöver jag inte vrida tillbaka huvudet manuellt.

Huvudvridningen är nu klar och sparad i ett lager som jag kan släcka ned när ifall jag inte önskar ha den längre.

3.2 Animationstest 2

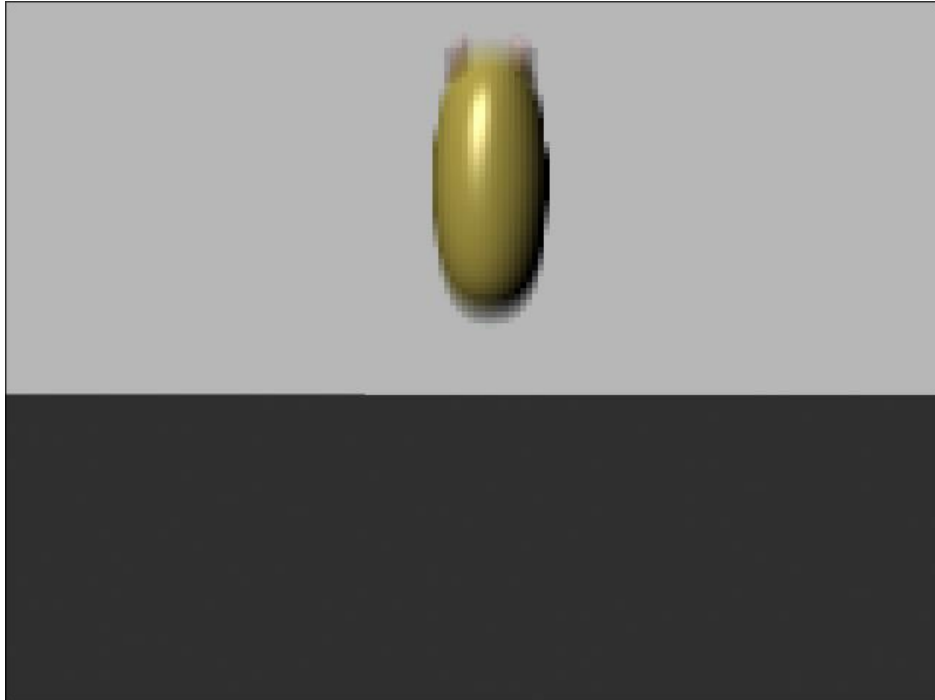
Det andra animationstestet (figur 8) jag gjorde var att en bollliknande karaktär (nedladdad gratisrigg[6]), som gör ett hopp. Med hjälp av trax-editorn och animationslager styr jag över hur högt detta hopp blir.



Figur 8. Bollkaraktären i det andra animationstestet. Här har den ursprung positionen.

Hoppet (figur 9) i animationen finns redan i grundanimationen. Det jag behövde verktygen till var med andra ord för att förlänga hoppet. Vid första anblicken av

uppgiften trodde jag att metoden för att förlänga en rörelse med hjälp av trax-editorn och animationslager skulle te sig annorlunda jämfört med hur man la till en rörelse (huvudvridning). Detta var inte fallet. Faktum är att jag hitta lösningen genom att använda nästan exakt samma metod som vid det första animationstestet.



Figurtext 9. Bollkaraktären när den är i luften efter att ha hoppat.

3.2.1 Animering 2 med trax-editorn

Jag sparade ned hela animationen inklusive hoppet i bollkaraktärens Character Set. Sedan satte jag keys för kontrollobjektet, som styr höjden för bollkaraktären, där Y-värdet börjar öka och när det är noll (När bollen lyfter från marken och när det landar). Vid den frame där Y-värdet hade sitt maxvärde, alltså högsta punkt för bollkaraktären, ökade jag värdet ytterligare och satte en key. Detta nya Y-värde anger högsta möjliga hopp för bollkaraktären. Jag sparar ned denna animation i ett klipp. När detta klipp är aktivt så hoppar bollkaraktären högre än vid originalet. Det går att ställa in värde för en funktion som kallas *Weight* för animationsklipp. Det avgör hur mycket av värdet som beskrivs i animationsklippet rörelsen ska ta av. Sänker man *Weight* så kommer karaktären att hoppa lägre.

3.2.2 Animering 2 med animationslager

Jag sparade ned hela animationen i ett eget lager. I ett annat lager satt jag keys precis likadant som vi experimentet jag beskrev i stycket ovanför. Animationslager har också en *Weight* funktion som det går att justera höjden på hoppet med.

4 Resultat och diskussion

Nedan står resultaten som erhållits av de olika animationstesterna och resonemanget bakom utfallet.

4.1 Resultat av animationstest

Resultaten av animationstesterna blev som följande:

4.1.1 Animationstest 1

Det var enkelt att skapa animationen för huvudvridningen både med hjälp av trax-editorn och animationslager. Båda verktygen gav ett riktigt bra resultat. Metoden går ut på att man lägger nya animationer på redan existerande animationer. Detta gör att så att rörelser från gångcykeln fortfarande finns kvar och påverkar även huvudvridningen, som till exempel att huvudet guppar, vilket ger ett bra och realistiskt resultat. En annan positiv egenskap är att gångcykeln i sig inte är förändrad så önskar man inte huvudvridningen längre gå bort den och gångcykeln är precis som den var innan.

Animationslager har den stora fördelen att det är väldigt lätt att ändra i animationen som man har sparat i ett lager. Detta beror på att alla keys för ett lager visas om det lagret är aktivt, på så sätt går det att ändra på dem väldigt enkelt (till exempel så att huvudet vrider sig mera eller mindre). Sparar man en animation i ett animationsklipp försvinner alla de keys från timeslidern och man måste ta långa omvägar för att låsa upp dem så att det går att ändra dem.

Fördelen med trax-editorn är att det är enklare att ändra på huvudvridningens helhet (startpunkt och hastighet, etc.) eftersom animationsklippet beskriver hela huvudvridningen medan keys bara är en del. Det är även enklare att kopiera huvudvridningen om den finns i ett animationsklipp än i ett animationslager, så att hjorten vrider på huvudet flera gånger.

4.1.2 Animationstest 2

För denna typ av animation fick *Weight*-funktionen stor användning. Med denna funktion behövs det bara ett lager eller animationsklipp för att styra precis hur högt hoppet skall vara. Det finns väldigt liten praktisk skillnad mellan att använda trax-editorn eller animationslager för denna typ av animation. Trax-editorn har fördelen att samma animationsklipp kan användas flera gånger ifall karaktären hoppar flera gånger. Använder man animationslager måste man antingen ändra i lagret efter hur många gånger karaktären hoppar genom att sätta keys på rätt ställen. Eller så skapar man egna lager för varje hopp, så att det går att styra höjden för alla hoppen enskilt.

4.2 Diskussion

Det är tydligt att trax-editorn och animationslager liknar varandra på många olika sätt. Den största skillnaden är att animationerna som sparas i animationslager inte sparas som klipp utan rakt av i timeslidern, vilket gör det betydligt enklare för animeraren att kunna ändra dess keyframes. I animationslager finns det heller inget krav att spara kontrollobjekten i Character Set. Animationslager är ett väldigt nytt

tillagt verktyg i Maya, medans trax-editorn har funnits längre. Animationslager är nästan som en uppgradering av trax-editorn.

Jag anser efter att ha jobbat mycket i båda de här verktygen att faktiskt inget av dem är en nödvändighet för att skapa animationer. Verktygen är bara metoder för att förenkla manipulationer av animationskurvorna i graf-editorn. Nästan, om inte, allt som går att göra i dessa verktyg går även att göra genom att dra eller att flytta keys i animationskurvorna, skala dem eller kopiera dem, etc. Verktygen är komplement för att förenkla arbetet. Möjligheten att lägga rörelserna i en animation i olika lager gör det mycket enklare för animeraren att ändra utvalda rörelser.

Största användningssområdet jag har hittat för dessa verktyg är när man vill lägga till en rörelse på en linjär och loopbar rörelse, såsom en gångcykel. Det är smidigt att ha en adderad rörelse till exempel till en gångcykel, i ett eget klipp eller lager som går att appliceras precis vid den tidpunkt man önskar. Detta utan att själva gångcykeln ändras från grunden. En annan viktig funktion vad gäller trax-editorn och animationslager är Weight-funktionen. Genom att ändra på bara ett värde så kan man styra över hur mycket av en rörelse, i ett klipp eller lager, ska uttryckas. Detta går inte att göra på ett smidigt sätt genom att själv förändra animationskurvorna manuellt, ifall rörelserna är komplicerade, alltså att de förändras i flera axlar.

Det svåra med att animera i Maya är inte att lära sig verktygen eller tekniken. Det som är nyckeln till en bra animation är att ta god tid på sig och inte nöja sig för enkelt. Man ska dock ha i bakhuvudet att det går att finslipa en animation i princip i en oändlighet innan den blir perfekt. Man måste hitta den gränsen där en animation är tillräckligt bra för ändamålet och att det därför inte längre är nödvändigt att spendera mer tid på att finslipa den. Det är även viktigt att bilda sig en god känsla för när en animation är bra och ser realistisk ut. Realistisk behöver förstås inte vara verklighetstrogen men rörelserna bör följa de lagar som finns i den världen där karaktären befinner sig i. Några viktiga punkter att tänka på är vart tyngdpunkten sitter och att varje rörelse oftast har en förberedelse av något slag. En bra känsla för animation får man främst genom att animera mycket, men också genom att studera rörelser överhuvudtaget. En effektiv metod för att förstå sig på en rörelse som man skall animera är att härma rörelsen själv med sig egen kropp. På så sätt märker man vart tyngden sitter och vilken kroppsdel som följer vad.

5 Slutsatser

- Vilka är de vanligaste metoderna för att framställa en animation?

Ifall det är en karaktär som skall animeras så riggas den först. Ett skelett tilldelas karaktären, olika typer av handtag kopplas till detta skelett. Handtagen och lederna i skelettet sammankopplas med kontrollobjekt som är de objekten som animeraren i slutändan flyttar när han animerar karaktären.

En enkel version av animationen görs först (som kallas *blocking* [8]) det går ut på att man sätter karaktären i animationens nyckelpositioner, utan att riktigt ta hänsyn till timing, etc. Efter att animeraren är nöjd med sin *blocking* så animerar han så att de interpolerande rörelserna mellan nyckelpositionerna ser bra ut.

- Vilka verktyg behöver en animerare för att kunna animera?

Det verktyget som är en nödvändighet för en animerare, är Graf-editorn. Det är i detta verktyg man skapar själva animationen. Det absolut vanligaste arbetssättet för animerare är att de först skapar en grund för animationen genom att keya positioner och poser. Sedan finslipar man timingen och de interpolerande rörelserna mellan dessa keys genom att ändra i kurvorna i graf-editorn. Detta sker bland annat genom att lägga ut mer keys för att ge större kontroll över graferna.

I princip all animering som går att göra i trax-editorn och animationslager går även att göra i graf-editorn genom att ändra dess kurvor. Trax-editorn och animationslager är med andra ord bara olika vägar eller hjälpmedel för att uppnå ett visst resultat.

- Vad har trax-editorn att erbjuda?

Med traxeditorn kan man spara ned animationer för ett eller flera kontrollobjekt i klipp. Dessa klipp går bland annat att flytta, skala och kopiera. På så sätt ges animeraren en effektiv och enkel kontroll över när animationerna skall starta, hur snabbt de skall spela och om de skall upprepas. Flera kontrollobjekt går att spara ned i s.k. Character Set för snabb åtkomst. Ett Character Set har ett eget bibliotek med animationsklipp som är sparade. Det går att spela flera animationsklipp på samma gång eftersom de ligger i olika spår, på så sätt kan man addera rörelser till andra rörelser, som till exempel en vinkande arm. Ett animationsklipp kan även innehålla en förstärkning av ett annat animationsklipp, alltså ett högre värde på en variabel som används i en annan animation, resultatet kan vara till exempel högre höjd eller längre steg för en karaktär.

- Vad har animationslager att erbjuda?

Med animationslager går det att placera animationer för en eller flera kontrollobjekt i lager. Animationerna i lagren går att applicera på redan existerande animationer för att utöka eller förändra animationer. Det är enkelt att stänga ned ett lager ifall man inte längre önskar animationen den innehåller. Det går även med animationslager att förstärka en redan existerande animation. Ett så kallat *override-lager* går att skapa med animationslager för att ta bort all rörelse på valda kontrollobjekt, men rörelserna finns fortfarande sparade ifall man önskar att använda dem igen.

5.1 Fortsatt arbete

För att utreda användningsområdena för animationslager och trax-editorn ytterligare kan man testa att lägga in animationer i en spelmotor. Genom att göra så skulle man kunna undersöka ifall spelmotorn fungerar bättre med animationerna i ett animationsklipp, animationslager, eller bara som kurvor i timeslidern. Ett annat område inom dessa verktyg som jag inte hann ta upp i detta arbete är partikeleffekter. Det går nämligen spara ned partikeleffekters rörelser i trax-editorn. Detta är en bra metod för att kontrollera partiklarnas rörelser. Det går till exempel ha en rörelse som gör att partiklarna flyger åt alla håll i ett animationsklipp. Kopierar man detta klipp och lägger det efter ett annat så kan man få effekten att partiklarna pulserar. Detta är bara några exempel som man kan ha som fortsatt arbete.

Referenser

- [1] <http://en.wikipedia.org/wiki/Animation> (2009-02-12)
- [2] http://www.imanishi.com/mayablog_en/2008/10/mayacharakter-setup-3-ik-handl.html (2009-02-12)
- [3] Lasseter, J. (1987). Principles of Traditional Animation Applied to 3D Computer Animation. Computer graphics, sid. 35-44 (SIGGRAPH '87).
http://www.siggraph.org/education/materials/HyperGraph/animation/character_animation/principles/prin_trad_anim.htm (2009-02-24)
- [4] <http://www.animationtoolworks.com/library/article9.html>. (2009-02-24)
- [5] http://www.siggraph.org/education/materials/HyperGraph/animation/rick_parent/Outline.html (2009-05-20)
- [6] http://www.highend3d.com/maya/downloads/character_rigs/download-5253.html?loc=ball_char_v01.rar (2009-05-12)
- [7] Paries, J. *Animations:Master Handbook*. S261.
- [8] Paries, J. *Animations:Master Handbook*. S6.