

Tiina Pyörälä

PELIHAHMON 3D-MALLINNUS

PELIHAHMON 3D-MALLINNUS

Tiina Pyörälä
Opinnäytetyö
Syksy 2015
Tietojenkäsittelyn koulutusohjelma
Oulun ammattikorkeakoulu

TIIVISTELMÄ

Oulun ammattikorkeakoulu
Tietojenkäsittelyn koulutusohjelma
Internet-palvelut ja digitaalinen media

Tekijä(t): Tiina Pyörälä

Opinnäytetyön nimi: Pelihahmon 3D-mallinnus

Työn ohjaaja: Teppo Räisänen

Työn valmistumislukukausi- ja vuosi: Syksy 2015

Sivumäärä: 49

Opinnäytetyön taustalla vaikuttajana oli kiinnostus pelikehitykseen. Mallintaminen on oma erikoisalansa, joka vaatii tekijältään vahvaa teoriapohjaa sekä pitkäjänteistä työskentelykykyä. Pelikehityksen graafisen toteutuksen vaiheissa tulee vastaan useamman osa-alueen käytäntöä, jonka johdosta tämä opinnäyte kattaa yleiskatsauksen myös mallinnusprosessin ulkopuolelle jäävistä termistöistä. Opinnäytteessä kerrotaan objektin rakenneosista, niiden muotoilutekniikoista sekä väritysprosesseista. Teoria kattaa myös runkomekanismin peruspilarit sekä katsauksen animaatioon. Mallinnustekniikat käsitellään omana osionaan, sillä mallinnustyöhön liittyen, ei ole yhtä ainoaa oikeaa toteutustapaa sen luomiseksi. 3D osajista on pulaa alan erikoistumisvaatimusten vuoksi, ja opinnäytteen tavoite oli luoda tiivis paketti mitä teoriaa olisi tärkeää hallita mallinnusuraa suunnitteleville.

Opinnäytteen tietoperustaa on sovellettu käytännöntoteutuksen kuvauksessa. Ymmärrys objektin rakenneosista sekä miten objektit liikkuvat ja muotoutuvat 3D avaruudessa oli avaintekijä mallinnustyön toteutuksessa. Tutkimusmenetelminä käytettiin tiedonhakua useammista lähteistä sekä käytännöntestauksen toteutuksia, määrittämään mitkä tekniikat tuottavat onnistuneimman tuloksen. Käytettyinä aineistoina pääosin olivat internet-lähteet aihealueen luonnon vuoksi. Kirjalähteenä käytettiin Vaughanin Digital Modeling teosta, joka keskittää suurimmalta osalta 3D mallintajan työnkuvan ja tarvittavat taidot.

Päätuloksena toteutettiin teoriaa hyödyntäen hahmomallinnustyö. Työn jatkokehityksenä on suunnitteilla soveltaa sitä toimivaksi osaksi pelimoottoriympäristöön, ja työn loppupäässä nähdään kevyt katsaus hahmomallinnuksen toteutuksesta testiversiossa.

Asiasanat: 3D-mallinnus, Mallinnustekniikat, Blender

ABSTRACT

Oulu University of Applied Sciences
Degree Programme of Business Information Systems
Option of Internet Services and Digital Media

Author(s): Tiina Pyörälä

Title of thesis: 3D modeling of a game character

Supervisor(s): Teppo Räisänen

Term and year when the thesis was submitted: Autumn 2015 Number of pages: 49

The background of the thesis was an influential interest in game development. Modelling has its own specialty, which also requires a strong theoretical basis as well as long-term working ability. Stages of the graphic realization of Game development will face practical of several sub-regions, which led to this thesis cover the overview of areas excluded from the modeling process terminologies. The thesis describes the structural components of an object, their design techniques and coloring processes. The theory also covers the fundamental pillars of the rigging mechanisms, as well as an overview of the animation. 3D talent is in short supply due to specialization in the sector, and the thesis objective was to create a compact package of the theory that is important to manage for those planning modeling career.

Thesis the knowledge base is applied to the practical implementation of the description. Understanding the structural components of an object as well as how objects move and take shape in 3D space was a key factor in the implementation of the modeling work. The research methods used were to search theory from a number of sources of information and practical testing of implementations, to determine which techniques produce the most successful results. The materials used were mainly internet sources, due to the nature of the industry. The book source was Vaughan Digital Modeling book, which concentrates most of the theory required in 3D modelers job description and the necessary skills needed.

The main result of the project, character modeling work was carried out by utilizing the theory. Further development for the project is planned to apply it as a functional part to the game engine environment, and at the end result of the project is seen as a light overview of character implementation to the test version.

Keywords: 3D modeling, Modeling techniques, Blender

SISÄLLYS

1	JOHDANTO	7
2	TIETOPERUSTA 3D MALLINNUKSELLE	8
2.1	3D ympäristön näkymä ja koordinaatistot	8
2.2	Objektin rakenne	10
2.2.1	Verteksi (vertex) , Pinta (face), Reuna (edge)	10
2.2.2	Normaalit (Normals)	10
2.2.3	Tyhjiö (Empty)	11
2.2.4	Hierarkiat	11
2.3	Muotoilu (Transforms)	13
2.3.1	Sijainnin muutos (Translation) ja koon muutos (Scale)	13
2.3.2	Kierto (Orientation, Rotation)	13
2.4	Muokkaimet (Modifiers)	15
2.5	Materiaalit ja tekstuurit	15
2.5.1	Shaders	15
2.5.2	Tekstuurikartat (Texture map)	16
2.5.3	UV kartoitus	17
2.6	Runkomekanismi (Rig)	18
2.6.1	Runko (Armature)	18
2.6.2	Meshin liittäminen (Skinning) ja asennon määrittäminen (Posing)	19
2.7	Animaatio	20
2.7.1	Kehys (frame) ja avainkehys (keyframe)	20
3	MALLINNUSTEKNIIKAT	21
3.1	Primitiivimallinnus	21
3.2	NURBS mallinnus	22
3.3	Polygonmallinnus	23
3.4	Subdivision mallinnus	24
4	MALLINNUSOHJELMISTON KUVAUS	25
4.1	Editorit	26
4.1.1	3D-editori	26
4.1.2	Animaatioeditori	27
4.2	Ominaisuudet -työkalurivi	28

4.3	Outliner.....	29
5	MALLINNUKSEN TOTEUTUKSEN KUVAUS.....	31
5.1	Rakenteen toteutus	31
5.1.1	Mallinnustekniikoiden vertailu	31
5.1.2	Siipien mallinnus	34
5.1.3	Mallinnuksen viimeistely.....	35
5.2	UV saumoitus	37
5.3	Materiaalien ja tekstuurien lisääminen	39
5.4	Runkomekanismin luonti	41
5.4.1	Luiden hierarkiat	41
5.4.2	Painomaalaus	42
5.5	Animaation toteutus.....	44
6	YHTEENVETO	46
7	LÄHTEET	48

1 JOHDANTO

Peliala on aktiivisesti kasvava ja kehittyvä ala. Laitteistojen tehokkuudet kasvavat teknologian kehityksen myötä, jonka johdosta voidaan tuottaa visuaalisesti toistaan kauniimpia pelejä. Pelit kehittyvät graafisesti yhä yksityiskohtaisemmiksi, sisällyttäen monimutkaisia toteutuskuvia ja suuria työryhmiä tuotoksien takana. Peligrafiikan tuottamisen voidaan lajitella kategorioihin; konseptitaide, mallinnus ja animaatio. Jokainen näistä kategorioista sisältää syvän tason tietämystä eri tekniikoista, työkaluista sekä kunkin alan teoriasta.

Mallintajan näkökulmasta peligrafiikkaa käsitellään sen teknisiltä osilta, keskittyen konseptitöiden toteuttamiseen 2D- sekä 3D-ympäristöihin. 3D-grafiikassa on huomioitava käyttöresursseja, optimoitu grafiikkatuotosten yksityiskohtien luonnit mahdollisimman keveiksi laitteistoille ja keskittyä mallinnettavien kohteiden rakenteen realistiseen ulkomuotoon. Mallintajan on pystyttävä luomaan hahmoja ja objekteja, joilla on myös toimivat mekanismit, jotta animaattorien ei tarvitsisi keskittyä perusliikekuvien (hahmon kävely, pallon pyöriminen) animaatioon.

Työn lähtökohdaksi oli omaehtoinen kiinnostus alaan ja sen tavoitteena oli kehittää henkilökohtaista osaamista ja luoda tarkempi käsitys 3D mallinnukseen liittyvistä konsepteista. Opinnäytetyön lopputuloksena on esitelty valmis mallinnustyö, jossa on sovellettu opinnäytteessä esiteltyjä tekniikoita sekä alan käsitteitä käytännötoteutuksina. Painotettuna, tässä työssä tarkastellaan 3D mallinnusta käsitellen sen termistöä ja siihen ei liity yhteensopivuutta pelimoottorien tai muiden kehitysympäristöjen kanssa. Kehitysympäristöjä sivutaan aiheena vain siinä määrin, kun mallinnuskohdetta demonstroidaan pelimoottoriympäristössä. Opinnäyte sisältää myös kuvauksen mallinnusohjelmiston käyttöliittymästä ja sen toiminnoista. Osio on sisällytetty opinnäytteeseen, sillä käyttöliittymän eri toimintojen ja mahdollisuuksien ymmärtäminen mallinnusprosessin yhteydessä on tärkeitä optimaaliseen lopputulokseen.

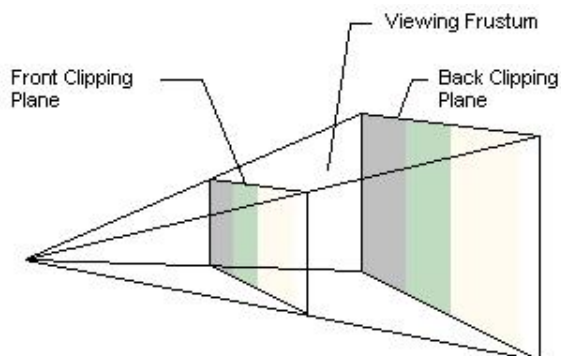
2 TIETOPERUSTA 3D MALLINNUKSELLE

Tässä luvussa käsitellään ensin 3D ympäristön rakennetta sekä keskitytään objektin rakentamiseen ja sen kokoonpanoon syvemmällä tasolla. Kappale etenee objektin muotoiluun liittyviin muuttujiin, luoden perustaa objektin käyttäytymiselle 3D maailmassa. Seuraavana vuorossa on kevyt katsaus muokkaimiin, joilla muutetaan objektin pintageometriaa. Tämän osion olen tarkoituksella jättänyt tiiviiksi, sillä mallinnusta toteuttaessa hyödynsin vain kolmea muokkainta. Luvussa esitellään materiaalit ja tekstuurit, eli se pyrkii avaamaan objektin väritykseen ja ulkomuodon varjostukseen liittyviä termejä. Runkomekanismiin ja animaatioon siirrytään luvun loppupäässä, jossa käydään läpi niiden terminologia sekä tuotokseen vaikuttavat osat.

2.1 3D ympäristön näkymä ja koordinaatistot

Näkymäkartio (view frustum) on 3D alue joka määrittää miten mallinnukset piirtyvät kameralle. Näkymäkartion voi kuvata leikattuna pyramidina, jonka pääly- ja alaosa ovat määritetty kaukokseksi ja lähipaneeleina. Näiden kahden paneelin välille muodostuvaa tilaa voidaan renderöidä perspektiivinäkymäkamerassa. (Unity3d.com, viitattu 12.9.2015)

Ohjelmistojen rasiuksen vuoksi, näkymä ei ole loputon. Objekteja, jotka sijaitsevat kaukana kamerasta tai todella lähellä sitä, ei renderöidä näkymään. Perspektiivinäkymässä kameraa lähellä olevat objektit piirtyvät suurempina kuin kauempana sijaitsevat. (Microsoft, viitattu 12.9.2015)



KUVIO x. Näkymäkartio. (Microsoft, viitattu 12.9.2015)

Perspektiivinäkymän ydin on edustaa yhdensuuntaisia kolmiulotteisen näkymän reunoja kaksiulotteisen kuvan reunoilla, jotka eivät ole yhdensuuntaisia. Kun tämä toteutetaan oikein, tämä tuottaa lähimpien objektien näkymisen suurempina kuin kauempana sijaitsevat ja luo syvyyssilluusion. (Wikibooks.org, viitattu 10.10.2015) Ihmissilmä on tottunut perspektiivinäkymään, jossa näemme objektit niiden etäisyyksien perusteella. Orthografinen näkymä vaatii totuttautumisaikansa, sillä objektit pysyvät samankokoisina etäisyydestä riippumattomana. Orthograafinen näkymä on yleisesti käytössä oletuksena 3D ohjelmistoissa, sillä se antaa teknisesti paremman näkökulman piirtämiseen ja sijaintien arviointiin. Näkymän kuvatyypin vaihtaminen ei muuta renderöinnin lopputulosta. (Blender Reference Manual 2015, viitattu 12.9.2015)

Orthograafinen näkymä 3D näkymästä on kaksiulotteinen kuva, jossa rinnakkaiset sivut näkyvät rinnakkain. Toisin sanoen kaikki reunat, jotka ovat kohtisuorassa näkymään suuntaan, esiintyvät suhteessa toisiinsa samaan kokoisina. Useimmiten tämänkaltaiset näkymät ovat linjassa pääakseleiden kanssa. Reunat, jotka ovat yhdensuuntaisia näkymän akselin kanssa, katoavat, ja ne jotka ovat rinnakkaisia muihin pääakseleihin näkyvät vaaka- tai pystysuoraan. Yleisimmin käytetyt orthograafiset näkymät ovat etu-, sivu-, ja ylänäkymät. Myös taka- ja pohjanäkymät ovat mahdollisia. (Wikibooks.org, viitattu 10.10.2015)

Koordinaatistoja on vasemmalle- ja oikealle kädelle. Vasemman käden x-akseli osoittaa oikealle, y-akseli ylös ja z-akseli pois päin käyttäjältä. Oikean käden akselit toimivat samalla periaatteella, mutta z-akseli osoittaa käyttäjää päin. Tästä syystä on mahdotonta asettaa näiden kahden koordinaatistojen z-akselien positiiviset arvot asettumaan samaan riviin. (Jamie King, viitattu 12.9.2015) Objekti voi olla liitännäisissä maailman-, objektin tai hierarkian koordinaatistoihin. Maailmakoordinaatistossa objektit perivät muutoksensa perustuen määriteltyyn pisteeseen. Objektin omassa koordinaatistossa muutokset perustuvat objektin omaan alkuperään. Alkuperä on yleensä objektin keskipisteessä, tai juuri sen alapuolella. Hierarkiaan perustuvassa koordinaatistoissa objektiin tapahtuvat muutokset periytyvät ylimmällä tasolla olevan objektin mukaan. (Blender Reference Manual 2015, viitattu 12.9.2015)

2.2 Objektin rakenne

Objekteissa kokonaisuus rakennetaan kolmesta rakenne-elementistä; vertekseistä, reunoista ja pinnoista. Objekti koostuu, kun useat pinnat yhdistetään toisiinsa jatkuvana kokonaisuutena. Seuraavassa käsitellään objektin rakennetta syvemmällä tasolla.

2.2.1 Verteksi (vertex) , Pinta (face), Reuna (edge)

Pinta on perusosa 3D polygonia. Kun kolme tai enemmän reunoja on yhdistettynä, tyhjää tilaa niiden välissä joka täyttyy näkyvänä kutsutaan pinnaksi. Pinnat ovat alueita, joihin lisätään shading materiaali. Verteksi on kiteytettynä yksi piste tai sijainti 3D avaruudessa. Se on useimmiten näkymätön renderöidessä. Reuna yhdistää kaksi verteksiä toisiinsa luoden niiden välille suoran viivan. Reunat ovat rautalankoja, jotka ovat näkyvillä, kun objektia tarkastellaan rautalankanäkymässä. (Blender Reference Manual 2015, viitattu 8.9.2015)

Alue, joka on kolmen tai neljän verteksin välillä, rajattuna reunoilla, ei välttämättä tarvitse sisältää pintaa. Jos tällainen alue ei sisällä pintaa, se näkyy läpinäkyvänä tai olemattomana renderöidessä kuvassa. (The Guerrilla CG Project, viitattu 3.11.2015)

2.2.2 Normaalit (Normals)

Polygonin pinta muodostuu normaaleista. Normaalit osoittavat siihen suuntaan, mihin polygonin pinta on suunnattu. Normaalit sallivat ohjelmiston laskevan polygonin varjostuksen. Muuntamalla pinnan verteksin suuntaa, voidaan määrittää polygonien normaalien suunnat. Polygon voi olla yksipuolinen tai kaksipuolinen, josta yksipuolisena se on toiselta puolelta näkymätön. (The Guerrilla CG Project, viitattu 3.11.2015)

Normaalit osoittavat siihen suuntaan, mihin polygonin pinta on kasvotusten. Verteksin normaalin suunta vaikuttaa pinnan liukuväriarjostukseen. Kun verteksin osoitussuuntaa käännetään, varjostus verteksin ympärillä muuttuu aiheuttaen liukuväriyypin varjostuksen. Verteksin näkökulmasta varjostus on kuin käännettäisiin tasaista polygonipintaa. Kun tarkastellaan kulmaa, jossa on terävä varjostus, voidaan kulman muodostavan kahden polygonin verteksi normaalin suuntaa

muuttaa siten että ne ovat samansuuntaiset, jolloin terävä reuna tulee näkymättömäksi. Tämä johtuu siitä, että molempien polygonien varjostukset ovat nyt yhdensuuntaiset. (The Guerrilla CG Project, viitattu 3.11.2015)

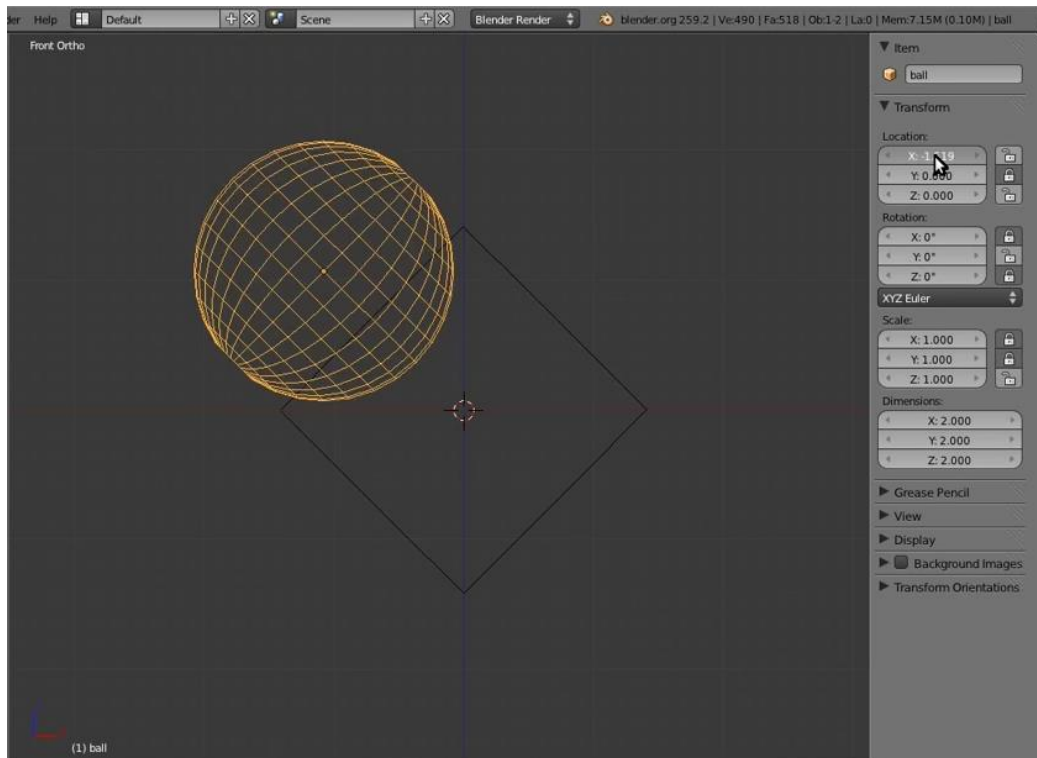
2.2.3 Tyhjiö (Empty)

Jokaisella näkymän objektilla on niiden muotoilun liittyvät asetuksensa, kuten sijainti näkymässä sekä kokonaismääräinen kierto ja koon muutos. Suurimmalla osalla objekteista on myös lisäasetuksia, kuten objektin oma geometria ja sen renderöinnin lopputuloksessa näkyvät materiaalit. Tyhjiöllä ei ole mitään näitä ominaisuuksia, eikä se näy renderöidessä. (Wikibooks.org, viitattu 11.10.2015)

Tyhjiö on objekti, jossa ei ole mitään, ei meshiä, ei kaaria, ei materiaalia tai muuta. Se on vain sijainti, kierre ja mittakaava näkymässä. Tyhjiöt ovat siis tyhjiä objekteja jotka ovat yksinkertaisia visuaalisia muotoilusolmuja, jotka eivät renderöidy. Ne ovat käteviä objektien sijainnin ja liikkeiden kontrolloinnissa. (Vegdahl 2014, viitattu 16.9.2015) Objektin fysiikan näkökulmasta, tyhjiö voidaan asettaa lähteeksi voimakentille (esim. tuuli). Tyhjiötä voidaan hyödyntää myös vanhempana monille objekteille, jolloin ne liikkuvat yhtenäisenä, kun tyhjiön sijaintia muutetaan. Tällaisissa tilanteissa tyhjiön läpinäkyvyys renderöinnissä on etu, koska se voidaan sijoittaa mihin tahansa vaatimaan paikkaan, ilman että se luo ylimääräistä epäjärjestystä näkymän lopputuotokseen. (Wikibooks.org, viitattu 11.10.2015)

2.2.4 Hierarkiat

Objekteja luodessa ja tehdessä niistä yhden toimivan kokonaisuuden, voidaan käyttää lapsi-vanhempi suhteita. Tietokonegraafiikassa, kun objektien välille luodaan tällainen suhde, lapsi perii muutoksensa vanhemmalta. Jos vanhempaa liikutetaan, lapsi kulkee vanhemman mukana. Sama pätee objektien koon muutokseen, jos vanhempaa litistetään, lapsi litistyy. Lapsi voi kuitenkin liikkua itsenäisesti vanhemman avaruudessa. Lapsiobjektin mittakaava-, sijainti- ja kierto määrittyvät vanhemman avaruuden mukaan. Vanhemman koon muutos vaikuttaa lapsiobjektin akseleiden toimintaan. Jos vanhemman kokoa muutetaan z-akselilla, lapsen z-akselin vaikutusarvot muuttuvat koon muutoksen mukana. (Vegdahl 2014, viitattu 16.9.2015)



KUVIO 1. Lapsiobjektin liikkuminen vanhemman avaruudessa. (Vegdahl 2014, viitattu 16.9.2015)

Käytännössä tämä tarkoittaa sitä, että lapsen sijainnin muuttaminen z-akselilla näkyy maailman-koordinaatistossa pienemmällä vaikutuksella, sillä se laskee sijaintinsa vanhemman mittakaavan mukaan. Koonmuutoksesta seurauksena, verrattaessa lapsiobjektin z- ja x-akselien sijainnin muutoksen voimakkuutta käyttäen samoja arvoja, lapsiobjekti liikkuu vanhemman avaruudessa suuremmalla vaikutuksella x-akselilla, kuin z-akselilla. (Vegdahl 2014, viitattu 16.9.2015)

Lapsi-vanhempi suhteet voivat olla useampitasoisia. Objekti, joka on vanhemman avaruudessa, voi myös olla toisen objektin vanhempi. Objektien lapsi-vanhempi suhdetta, jossa objekti säilyttää sijaintinsa ja suuntansa perimällä ne toisesta objektista, voidaan esittää hierarkioina. Hierarkioita kuvataan näkymäkaavioilla, joka muistuttaa sukupuuta. Jos hierarkia olisi sukupuu, jokainen ihminen olisi yksittäinen solmunsä. Jokaisella solmulla on omat muunnosmatriisinsä, joka määrittää solmun sijainnin, suunnan ja mittakaavan riippuen sen vanhemmasta tai vanhemman koordinaatistosta. Käsitteellisesti muunnosmatriisi porrastaa solmun sijainnin, suunnan ja mittakaavan sen vanhemmasta. (Worldviz.com, viitattu 14.9.2015) Vanhemman sijainnin muuttaminen, pyörittäminen ja koon muutos useimmiten muuttaa näitä arvoja myös lapsisolmulla. Kuitenkin lapsisolmun muuttaminen näillä arvoilla ei tee muutosta vanhempaan. Toisin sanoen, vaikutusvallan suunta toimii vanhemmasta lapseen, mutta ei lapsesta vanhempaan. (Blender Reference Manual 2015, viitattu 14.9.2015)

2.3 Muotoilu (Transforms)

Tietokonegrafiikassa muotoilut (Transforms) ovat sijainteja, kierteitä ja mittakaavoja tai niiden yhdistelmiä. Seuraavassa käsitellään muotoilujen käsitteitä ja kohdataan mahdollisia ongelma-kohtia niitä soveltaessa.

2.3.1 Sijainnin muutos (Translation) ja koon muutos (Scale)

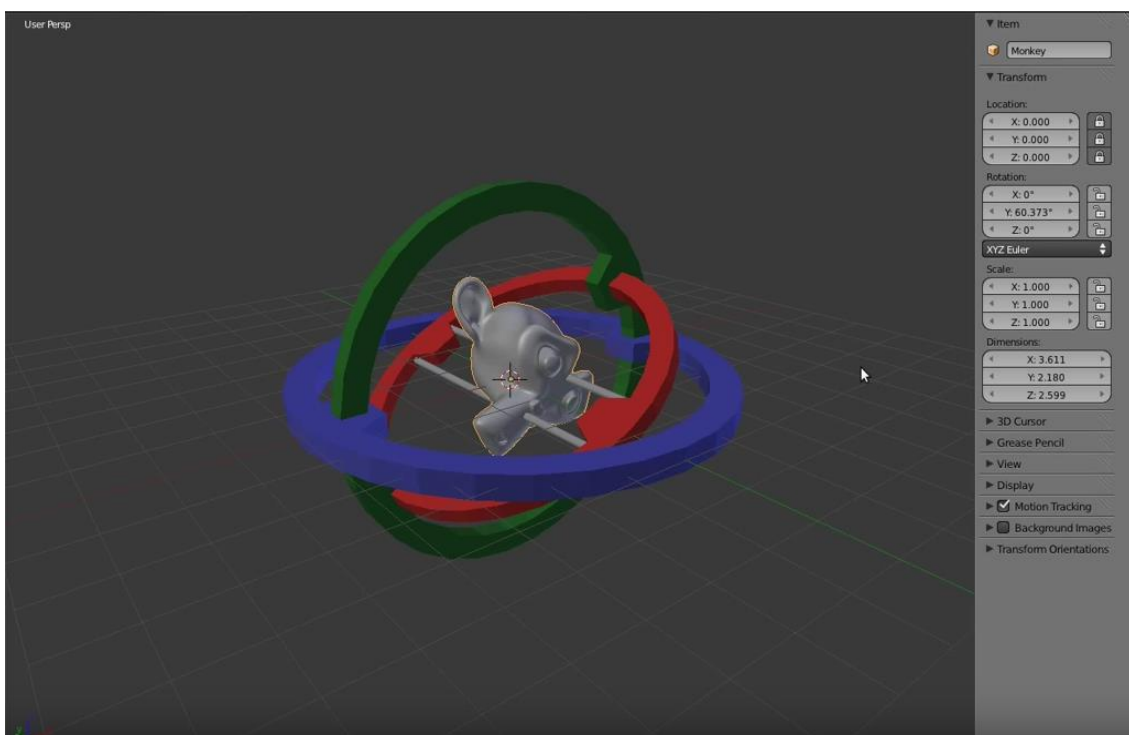
Translation tarkoittaa, että objektilla on muutettu sijainti. Koon muutosta tehdessä valittua elementtiä skaalataan sisään- tai ulospäin liitännäisissä hiiren osoittimen sijaintiin. Skaalauksen voimakkuus kasvaa sen mukaan, kun osoitinta siirretään kauemmas aloituspisteestä, ja laskee kun sitä siirretään lähemmäksi aloituspistettä. Jos osoitin ylittää aloituspisteen, objektin skaalaus jatkuu negatiivisilla arvoilla ja objekti kääntyy. (Wikibooks.org, viitattu 11.10.2015)

2.3.2 Kierto (Orientation, Rotation)

Kierron voi jakaa kahteen ryhmään, kiertoon (Rotation) sekä kierron sijaintiin (Orientation). Kierron sijainti tarkoittaa objektin asentoa kierron tekemisen jälkeen. Kierto itsessään tarkoittaa kierron muutosta. Jos objektia kierretään, lopputuloksena saadaan kierron sijainti, mutta toiminta itsessään on kierron muuttamista. Animaatioita tehdessä näiden kahden termin käyttämisen välinen ero on häilyväinen, sillä kierron muuttamisella määritetään kierron sijainti. Tätä voidaan tarkastella niin, että lähtökohtana objektilla on neutraali kierron sijainti, ja se halutaan asettaa erilaisiin kierron sijainteihin käyttämällä kierron muutosta. Animaattorit siis määrittävät kierron sijainteja hyödyntäen kierron muutosta. Kiertoa voidaan säätää käyttämällä Axis-anglea sekä Euler kiertoa. Axis-Anglen periaate toimii määrittämällä yhden akselin, ja pyörittämällä objektia akselin ympäri. Axis-angle käyttää 4 koordinaattia kierron määrittämiselle, 3 akseleille ja 1 kiertokulmalle. Axis-Anglella voidaan saavuttaa mikä tahansa kiertokulman objektille. Esimerkkinä puhuttaessa kahdesta kierron sijainnista 3D avaruudessa, on aina olemassa akseli, joka voidaan saavuttaa Axis-Anglella. (Vegdahl 2014, viitattu 16.9.2015)

Euler kierrot ovat pääosin kaikista tunnetuimmat kierron kuvaimet 3D animaatiopaketeissa. Pähkinänkuoressa, Euler kierrot ovat sarja 3 eri kiertoa, kukin ympäri pääakseleita. Esimerkiksi kier-

täessä objektia 20 astetta x-akselilla, 90astetta y-akselilla ja -40 astetta z-akselilla. Käyttäen kolmea pääakselia, voidaan saavuttaa mikä tahansa kierron sijainti objektille. Euler kierroilla kierrot tehdään järjestyksessä. Kauimpana sarjassa olevat kierrot vaikuttavat aikaisempien kiertojen akseliin. XYZ Euler kierron asettelussa, x-akseli on sisimpänä, y-akseli keskellä ja z-akseli ulommaisena. Kun y-akselia kierretään, x-akseli kulkee sen mukana. Sama pätee y-akselille ja z-akselille. Akseleita kierrettäessä voidaan todeta, että x-akseli, sekä y-akseli eivät ole rivissä maailman akseleiden kanssa, vaan objektin paikallisten akseleiden kanssa. (Vegdahl 2014, viitattu 16.9.2015)



KUVIO 2. Kierron akselit. (Vegdahl 2014, viitattu 16.9.2015)

Euler kierroissa akselien ryhmittymiset muuttuvat. Tämä voi aiheuttaa Gimbal lukon. Gimbal lukko on yleinen 3D kierroissa, jossa keskellä oleva akseli menee samaan riviin uloimman kanssa, jolloin menetetään yksi kierron akseli. Euler kierron järjestys ei ole pakotettu, ja on muunnettavissa. Eli mallintajan tiedostaessa, mitä akselia hän useimmiten käyttää, hän voi valita Euler kierroksi sellaisen asetelman, jossa Gimbal lukko on epätodennäköisin. Gimbal lukkoa voidaan myös välttää lukitsemalla yksi kolmesta akselistä, ja keskittää käyttö vain kahteen akseliin. (Vegdahl 2014, viitattu 16.9.2015)

2.4 Muokkaimet (Modifiers)

Muokkaimet ovat automaattisia toimintoja, jotka vaikuttavat objektiin rikkomatta sitä. Muokkaimien avulla voidaan suorittaa monia tehosteita automaattisesti, joka manuaalisesti toteutettuna veisi turhaa aikaa (esimerkiksi pintojen jakaminen) ja vaikuttamatta objektin pohjatopologiaan. Muokkaimet toimivat muuntamalla miten kohde näytetään ja renderöidään, muuttamatta objektin varsinaista geometriaa. Jos muutoksista halutaan pysyviä, voidaan ne lisätä objektiin. (Blender Reference Manual 2015, viitattu 17.9.2015)

2.5 Materiaalit ja tekstuurit

Oletuksena 3D objektilla on harmaa ulkomuoto. Lisäämällä shadereja mallinnus saa näyttävyyttä. Shaderit ovat nodeja, jotka vaikuttavat renderöinnin lopputuotokseen. Eri Shadereilla on ainutlaatuisia ulkomuotoja, ja niitä muokataan muuttamalla niiden arvoja (väri, läpinäkyvyys, kovuus, jne.) (Blender Reference Manual 2015, viitattu 11.10.2015)

Materiaalien asetukset tuottavat sileitä yhtenäisiä objekteja, jotka eivät ole lähellä realismia, jossa yhtenäisyys on harvinaista ja epätavallista. Käsitelläkseen tätä epärealistista yhtenäisyyttä, mallintaja voi soveltaa tekstuureja, jotka voivat muuttaa objektin heijastuvuutta, karheutta ja muita pintojen ominaisuuksia. Tekstuureilla on kolme pääkategoriaa; kuvat, proceduraaliset tekstuurit, jotka muodostuvat matemaattisilla kaavoilla, sekä ympäristökartat, joita käytetään heijastumisien ja taittojen vaikutelmien luomiseksi. (Blender Reference Manual 2015, viitattu 11.10.2015)

2.5.1 Shaders

Shaders ovat pieniä skriptejä, jotka sisältävät matemaattisia laskelmia ja algoritmeja kunkin pikselin laskemiseksi renderöintituotoksessa. Ne perustuvat valotukseen ja materiaaliasetuksiin. Materiaali määrittää käytetyt shaderit, ja valitut shaderit määrittävät mitkä asetukset ovat saatavilla materiaaleille. (Unity 2015, viitattu 11.10.2015)

2.5.2 Tekstuurikartat (Texture map)

Tekstuurikartat ovat lisäkerroksia pohjamateriaalin päälle, vaikuttaen yhteen tai useampaan objektin osan väriin. Kokonaisväritys koostuu tekstuurikarttojen vaikutusten kerrostuksista. Useamman kuin yhden tekstuurin käyttämistä hyödynnetään saavuttamaan yhdistetty vaikutus, jotta luoduista objekteista saadaan realistisen näköisiä. (Blender Reference Manual 2015, viitattu 11.10.2015)

Diffuusikartat (Diffuse map) määrittävät tekstuurin värimaailman. Pohjavärikartassa pitäisi olla varjoja vain vähän, tai ei lainkaan. Se ei kuitenkaan tarkoita, että se ei sisältäisi useita varjostustasoja, mutta kovat varjot tulisi määrittää vahvoilla nostetuilla reunoilla ja epämääräiset alueet jätetään huomiotta. Normaalikartoilla (Normal map) määritetään tekstuurien varjostukset kun valo osuu reunaan, ja kahden eri karttatyyppin käyttäminen varjostuksessa luo huonon lopputuloksen. Normaalikartat sallivat yksityiskohtien määrän korkeapolymalleissa matalan polyn vaatimalla väännöllä, jolloin renderointi vaatii vähemmän käytössä olevalta laitteistolta. Yksinkertaisuudessaan normaalikartat ovat hienojakoisia korkeuskarttoja. Poiketen yksinkertaisen harmaan kartan käyttämiseksi korkeuksien määrittelemisessä, normaalikartat käyttävät nelivärijärjestelmää määrittämään korkeuksien kaarevuutta. Tämän vuoksi voidaan saavuttaa tarkempi esitystapa valon vaikutukselle tekstuurin pinnassa. (Cain 2011, viitattu 11.10.2015)

Tekstuurin valotuksen heijastuvuutta voidaan määrittää kovuuskartoilla (Specular map). Kovuuskartat eivät heijasta ympäröivää maailmaa, vaan valon kovuutta. Yksinkertaisuudessaan kovuutta voidaan säätää valkoisella ja mustalla värillä, valkoisen edustaessa täyttä kovuutta ja mustan sen vastakohtaa. Heijastuskartat (Reflection map) antavat tekstuureille kiiltävän ulkomuodon. Fresnel on fyysisesti tarkempi heijastusratkaisu. Jyrkässä katselukulmassa (90 astetta) heijastukset voimakkuus vähenee, kun taas matalampi kulma korostaa heijastuksien voimakkuutta. (Cain 2011, viitattu 11.10.2015)

2.5.3 UV kartoitus

Tekstuuria lisätessä mallintajan on säädettävä objektin UV koordinaatteja, jottei se väännä tai veny alueille, mille sen ei ole tarkoitus. Tämä tehdään usein UV tekstuuri editorissa, missä mesh on levitettyä viitaten näkyvässä valittuun meshiin. Joustavin tapa levittää 2D tekstuuri 3D objektiin päälle on käyttää UV kartoitus tekniikkaa. Prosessissa otetaan kolmiulotteinen mesh (X, Y ja Z) ja levitetään tasaiselle kaksiulotteiselle kuvalle. Kuvan värit kartoittuvat meshin ympärille ja näkyvät meshin faceissa levitettyinä. UV teksturoinnilla voidaan luoda realistista ulkomuotoa objekteille tarkemmin kuin käyttämällä proseduraalisia materiaaleja ja tekstuureja, sekä verteksi maalausta. (Blender Reference Manual 2015, viitattu 9.10.2015)

UV kartoitus kuvaa mikä osa tekstuurista tulee olla liitettynä mihinkin polygoniin mallinnuksessa. Jokaiselle polygonin vertexille asetetaan 2D koordinaatit, jotka määrittävät mikä osa kuvasta levitetään. Näitä 2D koordinaatteja kutsutaan UV:eiksi, jotka ovat verrattavissa 3D koordinaatiston XYZ koordinaatteihin. Operaatiota UV kartoituksen luomisesta kutsutaan myös "Unwrap" termillä. Termi tulee käytännöstä, joka muistuttaa meshin levittämistä 2D tasoksi. Monimutkaisille mallinuksille tyypillinen kuutiomainen-, sylinterimäinen- tai pallokartoitus ei useimmiten riitä. Tarkempiin töihin voidaan käyttää saumoitustekniikkaa UV kartoituksen määrittelemiseksi. (Blender Reference Manual 2015, viitattu 9.10.2015) UV kartta on kaksiulotteinen ruudukko, joka koostuu pysty- (U) ja vaakasuuntaisista (V) koordinaateista. Koordinaatit kertovat miten kaksiulotteinen tekstuuri asetetaan kolmiulotteiseen objektiin. UV pisteet hallitsevat mitkä pisteet (pikselit) tekstuurissa vastaavat mitäkin pisteitä (vertex) meshissä. Jotta voidaan varmistaa, että UV kartoitus on oikein sovitettuna mallinnukseen, se tulee kartoittaa, jonka jälkeen manuaalisesti asetella se parhaimpaan mahdolliseen kokoonpanoon. (Modotutorials, viitattu 9.10.2015)

UV kartoituksen voi toteuttaa usealla tavalla. Esimerkkinä voidaan käyttää sylinterimäistä tai pallomaista kartoitusta. Sylinterimäinen kartoitus esittää UV:et virtuaalisen sylinterin muotoisen muodon läpi, joka on käärittynä objektin ympärille. Tämä projektisointitapa toimii parhaiten objekteilla, jotka ovat täysin suljettuja ja määriteltävissä sylintereinä. Sylinterikartoitus voi olla käytännöllinen kartoitustapa mallinnuksen osille, kuten orgaanisien mallien käsivarsille tai kovapintaisten meshien letkuille ja putkiosille. Pallomainen kartoitus esittää UV:et virtuaalisen pallomaisen muodon kautta, joka ympäröi objektin. Tämäntyyppinen kartoitus toimii parhaiten täysin suljettavissa palloa muistuttavissa objekteissa. (Modotutorials, viitattu 9.10.2015)

2.6 Runkomekanismi (Rig)

Mallintajan on säädettävä hahmolle tehtävässä runkomekanismissa vaatimuslista, mitä toimintoja varten se toteutetaan hahmolle. Tämä tarkoittaa sitä, että määritellään, mitä toimintoja ja millaisissa tilanteissa animaatiohahmo tulee esiintymään. Runkomekanismin tavoite on helpottaa ja nopeuttaa animaation tekijän työtä. Runkomekanismin suunnittelussa on hyvä keskittyä automatisoitujen toimintojen luomiseen, jotta animaation tekijän ei tarvitse suorittaa mallinnukselle ylimääräisiä kiertoja, mittakaava- tai sijaintimuutoksia. (Vegdahl 2014, viitattu 16.9.2015) Tämä on tärkeää, sillä hahmo, joka on esimerkiksi suunniteltu lentämään, ei välttämättä sopeudu heilumisanimaation luomiseen, jossa raajojen tulee roikkua hahmon perässä painovoiman mukaan. (The Guerrilla CG Project, viitattu 16.9.2015)

2.6.1 Runko (Armature)

Rungot muistuttavat tyhjiöitä. Ne eivät sisällä meshejä tai mitään, mikä on renderöitävissä. Ne sisältävät luuta, jotka ovat samankaltaisia kuin tyhjiöt. Luut ovat siis sijainteja, kierteitä ja mittakaavoja näkyvässä. (Vegdahl 2014, Viitattu 16.9.2015) Luut ovat runkojen peruselementtejä, ja ne koostuvat kolmesta osasta, juuresta, vartalosta ja kärjestä. Luun kärki ja juuri määrittävät sen sijainnin. Luut ovat jäykkiä elementtejä, jotka ovat valmistettu segmenteistä. Oletusarvoisesti jokaisella uudella luulla on vain yksi segmentti, joten sitä ei voi taivuttaa kaarevasti. (Blender Reference Manual 2015, viitattu 12.10.2015)

Runko on objektityyppi jota käytetään runkomekanismin luomiseen ja se ottaa vaikutteita reaali maailman luurakenteista. Rungot muodostuvat luista, jotka ovat oletusarvoisesti jäykkiä elementtejä. Luut rungon sisällä voivat olla täysin toisistaan riippumattomia, jolloin niihin tehdyt muotoilut eivät periydy. Tämä ei kuitenkaan ole yleinen ja käytännöllinen tapa luoda runkoja, sillä esimerkiksi jalan runkomekanismin luonnissa on toivottavaa, että reisiluuta seuraavat luut liikkuvat sen mukana. Luiden asettamisella hierarkiaan, voidaan luoda luuketjuja, jossa luiden kärjet ovat linkitettyinä lapsiluiden alkuperään. Kärki ja alkuperä voivat olla yhtenäisiä, jolloin ne ovat aina samassa pisteessä, tai ne voivat olla omina osinaan asetettuna hierarkiaan. Yksi luu voi toimia vanhempana useammalle lapsiluulle, jolloin se voi olla osana useampia luuketjuja samanaikaisesti. Luuta luuketjun alussa kutsutaan juuriluuksi, ja ketjun loppupään viimeistä luuta kärkiluuksi. (Blender Reference Manual 2015, viitattu 12.10.2015)

2.6.2 Meshin liittäminen (Skinning) ja asennon määrittäminen (Posing)

Rungot voidaan yhdistää kokonaisuksi objekteihin, jolloin voidaan vaikuttaa objektin muotoiluominaisuuksiin runkomekanismin kautta. Rungot voivat myös muokata objektin muotoa, vaikuttaen sen vertekseihin ja hallintapisteisiin. Tällöin lapsiobjekti on liitetty (skinned) koko runkoon, jolloin jokainen luista hallinnoi lapsiobjektin pintageometriaa. Luut voivat vaikuttaa objektin muotoon kahdella eri tavalla; ympäröintiprosessilla (Envelope) , joka käyttää läheisyys- ja vaikutusarvoja määrittämään, mitä objektin osia ne voivat muotoilla, sekä verteksiryhmillä (Vertex Groups), jolloin luun vaikutus ilmenee vain säädetyillä samannimisillä verteksiryhmillä, käyttäen verteksien painoarvoa säätöarvona. Jälkimmäinen toimintatapa on tarkempi, mutta vaatii enemmän aikaa, ja molempia tapoja voidaan käyttää samanaikaisesti. (Blender Reference Manual 2015, viitattu 12.10.2015)

Ympäröintiprosessi on yleisin liitostapa, sillä sitä voi käyttää useampien objektityyppien kanssa. Se perustuu luiden ja objektien geometrian läheisyysarvoihin, jossa jokaisella luulla on kaksi vaikutusaluetta, sisäinen sekä ulkoinen alue. Sisäinen alue, ilmenevänä luun kiinteänä osana, on hallittavissa luun kärjen ja juuren säteellä. Tällä alueella oleva objektin geometria on kokonaan luun hallinnassa. Ulkoisen alueen säteellä objektin geometria on vähemmän ja vähemmän liikkeen vaikutuksen alaisena luun siirtyessä kauemmaksi. Luun ominaisuus, painoarvo (weight), hallinnoi luun globaalia vaikutusta objektin muodon yli. Luu jolla on korkea painoarvo, on vaikutukseltaan voimakkaampi kuin luu, jolla on matala painoarvo. (Blender Reference Manual 2015, viitattu 12.10.2015)

Verteksiryhmä liittäminen voidaan käyttää vain objekteissa, joissa on verteksiryhmiä. Toimintatapa on yksinkertainen, jokaisen luun vaikutusvalta ylittää vain niihin vertekseihin, jotka kuuluvat luun nimen mukaan nimettyyn verteksiryhmään. Luun liikkeen vaikutusta verteksiin hallinnoi verteksin painoarvo tässä verteksiryhmässä. Verteksi voi kuulua useampaan eri verteksiryhmään, jolloin sen vaikutus ylittää useampaan luuhun, sillä voimakkuudella, mikä painoarvo sille on asetettu mitään luuta kohden. (Blender Reference Manual 2015, viitattu 12.10.2015)

Kun runko on liitetty vaadituista objekteista, voidaan aloittaa sen asennon määrittäminen. Muokkaamalla luita voidaan uudelleen muotoilla tai muuntaa objektien geometrian asentoa. Asennon määrittäminen tarkoittaa objektien muotoilua ja järjestelmistä tietyllä tavalla, jotta luodaan mielenkiintoinen kokoonpano. Asennon määrittämiä käytetään myös animaatioiden luonnissa. (Blender Reference Manual 2015, viitattu 12.10.2015)

2.7 Animaatio

Animaatio tarkoittaa objektille liikkeen luomista tai muodon muunnosta. Objekteja voidaan animoida usealla tavalla; liikuttamalla kokonaisuutena muuntaen sen sijaintia, kiertoa tai kokoa aikajanana edetessä, muodonmuutoksilla vaikuttaen vertekseihin ja hallintapisteisiin sekä periytymällä toisen objektin liikkeestä, esimerkkinä rungot. Animaatio tyypillisesti saavutetaan käyttämällä avainkehysjä. (Blender Reference Manual 2015, viitattu 22.11.2015)

2.7.1 Kehys (frame) ja avainkehys (keyframe)

Kehys on otos yhdestä näkymän hetkestä aikajanalla. Animaatio koostuu onnistuneesta sarjasta kehyksiä jotka edustavat peräkkäisiä hetkiä ajassa. Jos kehykset esitetään sujuvasti nopeana janana (24 kuvaa sekunnissa), silmää huijataan näkemään sujuvaa liikettä erinäisten asentojen peräkkäisyyden sijasta. (Wikibooks.org, viitattu 22.11.2015)

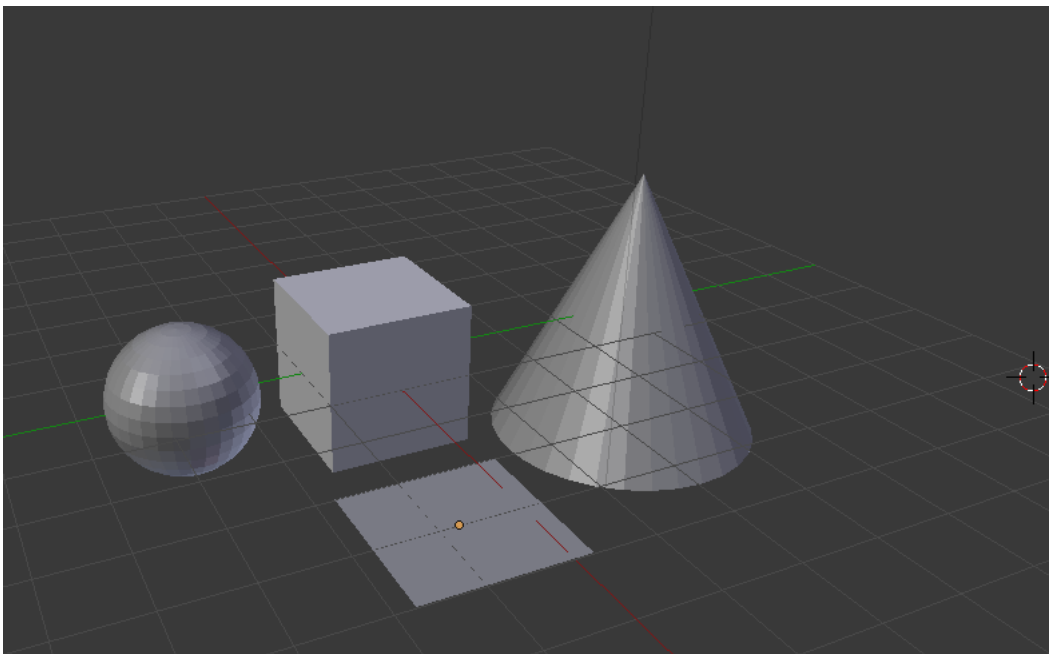
Keskeiset kohdat aikajanana animaatioissa ovat objektien sijainnit ja asennot. Ohjelmistolle määritetään näiden olevan avainkehysjä kyseisille muunnoksille (Sijainti, kierto, koon muutos) koskien valittua objektia. Kun animaatio suoritetaan, ohjelmisto interpoloi määritellyt muunnosparametrit avainkehysjien välillä, luoden tasaisen liikkeen asentojen välille. (Wikibooks.org, viitattu 22.11.2015)

3 MALLINNUSTEKNIIKAT

Luvussa 3 esitellään mallinnusprosessissa käytettäviä toteutustapoja. Luku käsittelee mallinnustekniikoita määrittellen niiden peruselementit, sekä esittelee konkreettisesti kuvien avulla tekniikoiden erot.

3.1 Primitiivimallinus

Primitiivimallinnus viittaa tyypilliseen mallinnustekniikkaan, jossa mallintaja käyttää työkaluohjelmiston tarjoamia primitiivimuotoja (kuutiot, spheret, jne.). Valmiita muotoja muokataan vastaamaan tarvittuja kriteereitä käyttäen ristikoita ja muokkaimia (taivutus, kierto jne.). (Asokan 2008, viitattu 20.8.2015)

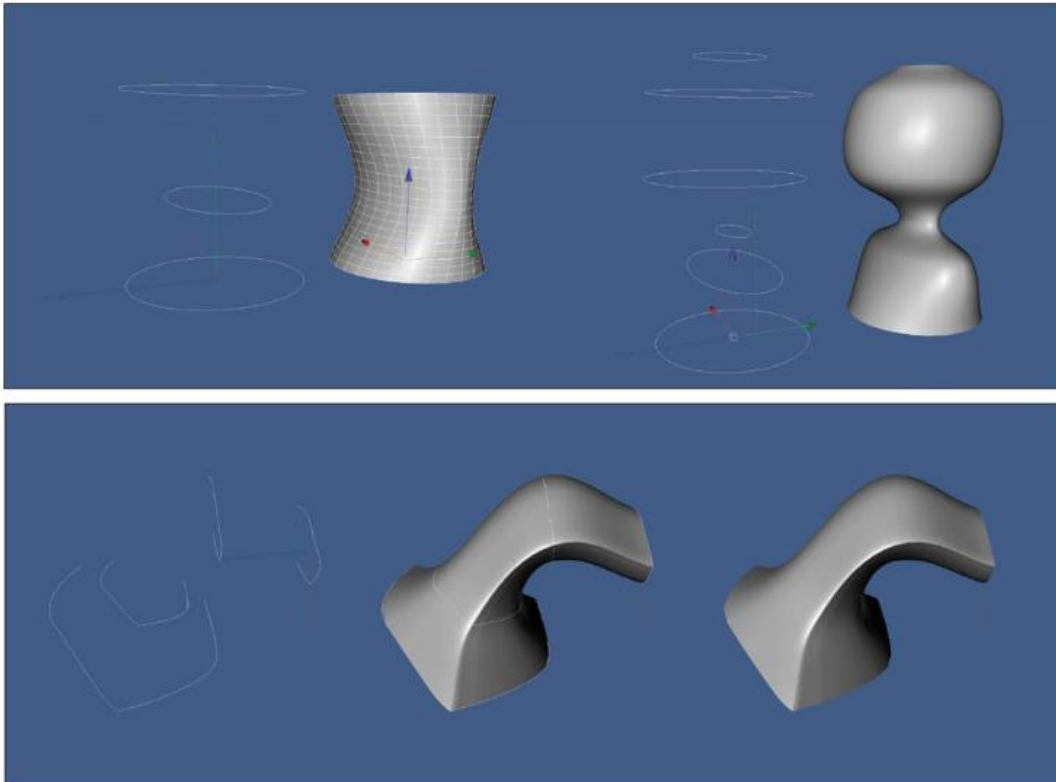


KUVIO 3. Primitiiviobjektit

Primitiivimallinnuksen lähtökohtana luodaan valmisprimitiivi, esimerkiksi kuutio, pallo, sylinteri tai kartio. Yhdistämällä näitä muotoja toisiinsa, luodaan mallinnuskokonaisuus, jotka lähdetään työstämään. Tekniikka on hyödyllisin mallinuksissa, joissa luotu kohde koostuu useista osista ja primitiivimuodoista. Haittapuolena on lopputuloksen ulkonäkö, josta selkeästi voidaan erottaa sen koostuvan useista primitiiviosista. (Axelsson 2013, viitattu 10.12.2015)

3.2 NURBS mallinnus

NURBS (Non-uniform rational basis spline) on matemaattinen käyrä, joka kuvaa yhtälöryhmää. Mallintajan näkökulmasta kyseessä on kokoelma käyriä, joiden hallintapisteillä voimme muokata mallia. (Asokan 2008, viitattu 20.8.2015)

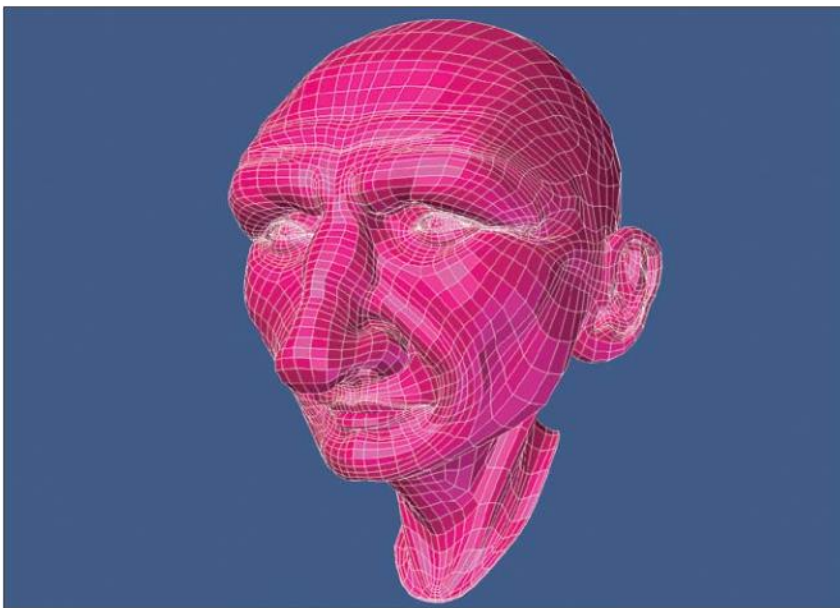


KUVIO 4. NURBS pinnat. (Vaughan 2012, viitattu 13.10.2015)

NURBS pinta on pehmenetty mesh, joka koostuu sarjasta splinejä (polynomisia kaaria). Tämä pehmenetty pinta muunnetaan renderöinnissä polygoneiksi, joten NURBS pinnat voivat sisältää satunnaisen määrän polygoneja. NURBSit voidaan muuntaa polygoneiksi tai jaetuiksi pinnoiksi (Subdivision surface), ja ne ovat käytännöllisiä useamman tyyppisten orgaanisien 3D muotojen rakennuksessa. NURBSien geometria on sileää oletusarvoisesti, eikä pintaa tarvitse jakaa pehennysvaikutuksen luomiseksi, toisin kuin polygon geometriassa. Sanat "Non-Uniform" viittaavat kurvien parametrien määrittämiseen. NURBS kurveja käytetään pääosin tietokoneavusteisessa suunnittelussa, teollisuudessa ja konetekniikassa. (Vaughan 2012, viitattu 12.10.2015)

3.3 Polygonmallinnus

Polygonit, joista yleisemmin käytetään nimitystä pinta (Face), ovat pisteistä koostuvia geometrisiä muotoja, jotka muodostavat 3D objektien pinnat. Polygon on useimmiten se osa, mikä on nähtävillä renderöintivaiheessa ja tyypillinen 3D mallinnus muodostuu sadoista ja tuhansista polygoneista. Vaikka useat mallinnusohjelmat sallivat yksi-, tai kaksipisteisiä polygoneja, yleisimmin käytössä ovat kolmipisteiset polygonit. (Vaughan 2012, viitattu 13.10.2015)



KUVIO 5. Polygoneista mallinnettu pää. (Vaughan 2012, viitattu 13.10.2015)

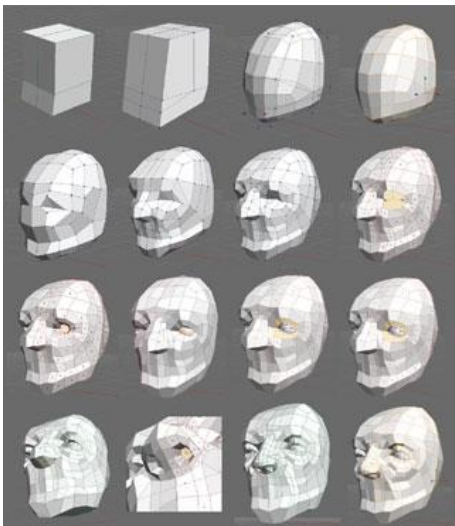
Polygonit ovat yksinkertaisimpia työstää ja niitä voidaan muokata vertexpisteitä, reunoja ja pintoja käyttäen. Objektin muotoa voidaan muuttaa käyttäen muokkaimia tai polygon työkaluja. Aloittaen joko valmiista primitive meshistä tai aivan puhtaalta pöydältä, voidaan työkaluilla leikata tai työntää ulospäin saadakseen objektille lopullisen muotonsa. Polygonien tulee olla yksinkertaisia, eikä niiden reunojen tule muodostaa päällekkäisyyksiä tai verteksien tule olla päällekkäin. (Blender Reference Manual 2015, viitattu 12.10.2015)

Polygonimallinnus on harvoja tekniikoita, joita käytetään useissa mallinnus-skenaarioissa, sekä orgaanisissa että epäorgaanisissa. Peruspohjana 3D mallintaja piirtää muodon käyttäen polygon työkalua, josta polygoneja lähdetään jakamaan pienempiin osiin ja työnnetään ulospäin, jotta saavutetaan lopullinen muoto. Mallinnus koostuu vain polygoneista, eikä muodon pinnat yleensä ole

pehmeitä ja tarkkoja. Prosessin jälkeen mallintaja lisää pehmentäviä muokkaimia, joka muuntaa mallin realistiseksi ja sileän näköiseksi hahmoksi. (Asokan 2008, viitattu 20.8.2015)

3.4 Subdivision mallinnus

Subdivit ovat yhdistelmä nurkseja ja polygoneja. Subdivien etu on, että ne sisältävät nurbsien pehmeiden, mutta sallivat yksityiskohtien muokkaamisen hallintapisteillä muuttamatta suuremmin meshin muotoa. Mallinnusprosessi lähtee liikkeelle polygon mallista, joka on luotu ja muunnettu subdivisional muodoksi. Muodolla on ristikoita ja hallintapisteitä, joilla mallintaja muokkaa alla olevaa alkuperäistä polygonmallia. Tämän johdosta alueille, jotka vaativat tarkempia yksityiskohtia, voidaan laittaa tiheämmin hallintapisteitä ja suuremman määrän verteksejä, toisen alueiden pysyessä alhaisemmissa tarkkuuksissa. (Asokan 2008, viitattu 20.8.2015)

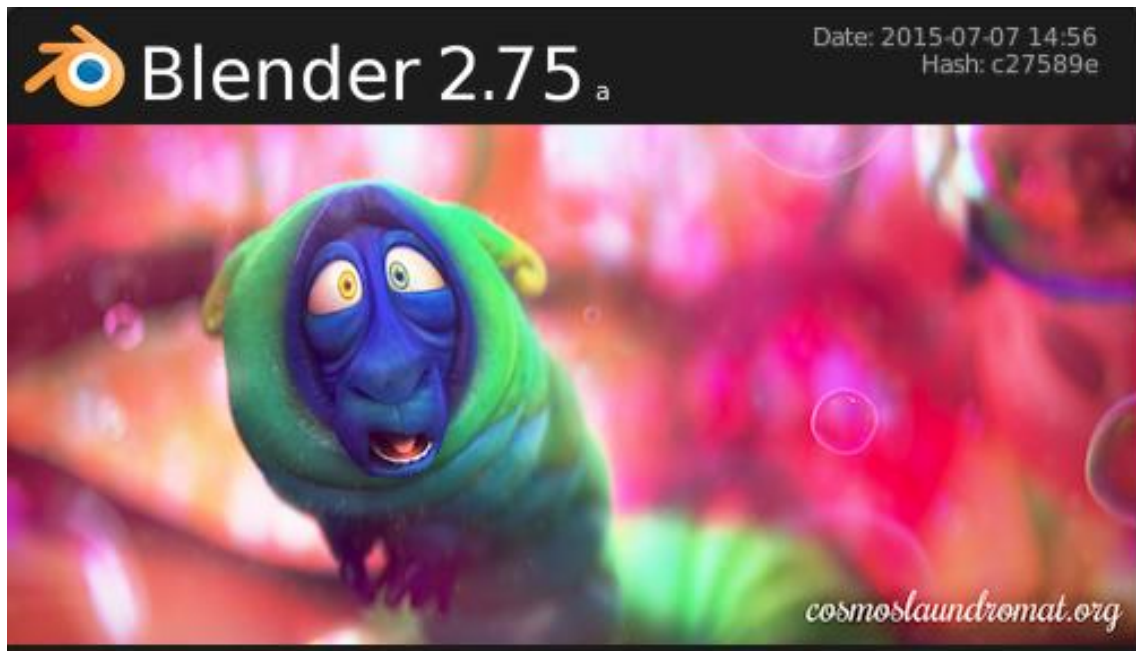


KUVIO 6. Subdivisionmallinnus (Holden 2011, viitattu 13.10.2015)

Subdivision on prosessi, joka pehmentää meshin pintoja lisäämällä polygonien ja verteksin määrää säilyttäen objektin alkuperäismuodon. Quad jakautuu neljään osaan myötäillen objektin ääri- viivoja. Tämä prosessi on käytössä useissa mallinnuksen toiminnoissa. Se sallii meshien rakentamisen hallittavissa olevilla polygonilukumäärillä. Quadit ovat ainoita pintaosia, jotka jakautuvat täydellisesti, jolloin on suositeltavaa käyttää quadeja meshien avainsijainneissa. Tekniikka sallii jakamisen voimakkuus levelien välisen vaihtelun. Näin mallintava voi esikatsella työtään ja nähdä miltä se näyttää pehmenettynä. Tekniikassa keskitytään ensisijaisesti meshin muodon ja vartalon määrittämiseen, josta jatketaan polygonien lisäämisellä yksityiskohtien ja syvennysvaikutusten luomiseen. (Holden 2011, viitattu 12.10.2015)

4 MALLINNUSOHJELMISTON KUVAUS

Toisessa luvussa käsitellään mallinnustyössä käytettyä ohjelmistoa. Mukaan on otettu vain ne osat ohjelmistosta, joita hyödynnetään työn toteutuksessa sekä luku sisältää kevyen katsauksen ohjelmiston historiaan.



Kuvio 7. Blender-ohjelmiston aloitusruutu ja ohjelmistoversio

Blender on avoimen lähdekoodin ohjelmisto, jota kehittävät maailmanlaajuisesti yksittäiset artistit, ammattilaiset ja harrastelijat, sekä opiskelijat, VFX -asiantuntijat, animaattorit, että pelisuunnittelijat. Heidän tavoitteensa on tehdä ilmainen työkalu 3D grafiikan luonnin työprosessia varten. (Blender.org 2015, hakupäivä 6.9.2014)

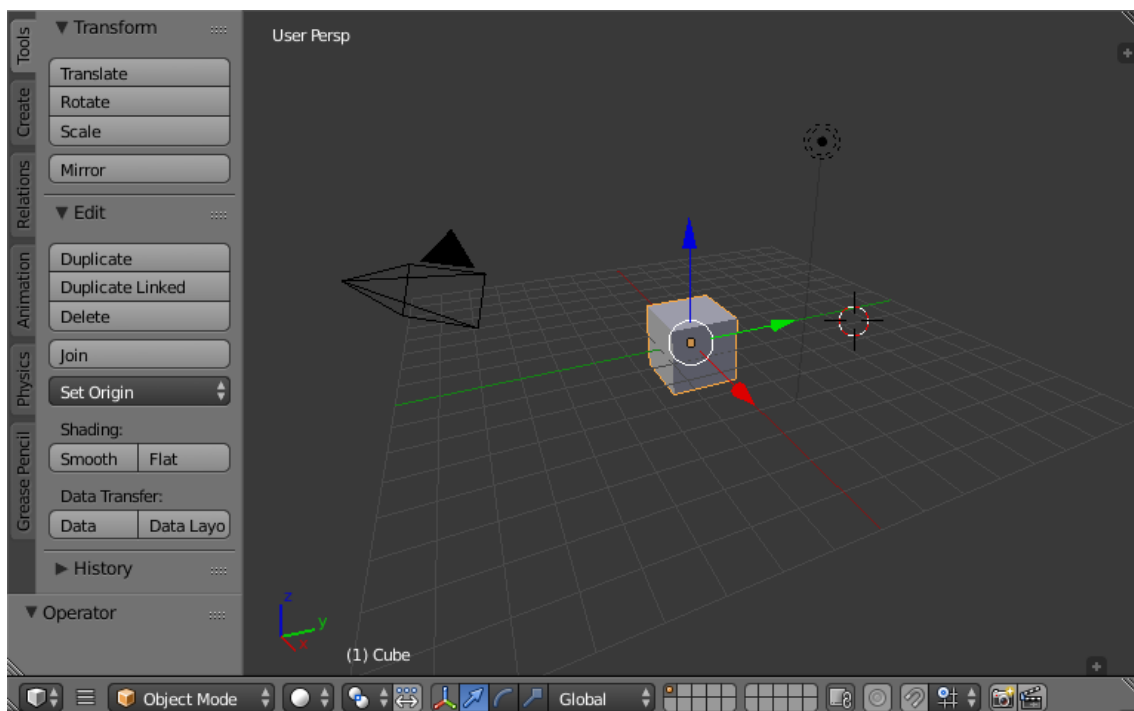
Blender -projektia kehitetään GNU GPL lisenssin alaisuudessa. Ohjelmisto on tuotettu C-, C++ ja Python-ohjelmointikielillä. Vapaaehtoiset ja ammattilaiset ottavat osaa virallisen julkaisun tuottamiseen. Tämä työryhmä sisältää kehittäjiä, skriptiaajia, kääntäjiä, suunnittelijoita sekä käyttäjiä, testaavat ohjelmistoa ja antavat palautetta sen toiminnallisuudesta. (Blender.org 2015, viitattu 6.9.2014)

4.1 Editorit

Seuraavassa tutustutaan Blender -ohjelmiston editoreihin ja niiden toimintoihin sekä työkaluihin. Kappaleessa käydään läpi työkalujen toiminnot sekä editorien eri esitystavat objektiin liittyvistä muutoksista. Kappale pyrkii luomaan kokonaiskuvan pääosaisista editoreista mallinnustyössä.

4.1.1 3D-editori

3D editoria käytetään 3D näkymän vuorovaikutuksessa eri tarkoituksiin, kuten mallinnukseen, animaatioiden tekemiseen ja tekstuurimaalaukseen. Navigointia voi näkymässä tehdä hiiren liikkeillä sekä näppäimistön pikanäppäimillä. (Blender.org 2015, viitattu 18.9.2015)



Kuvio 8. 3D-editori

Objekteja voi myös tarkastella paikallis- sekä globaalinäkymissä. Paikallinäkymässä käyttäjä eristää valitun objektin siten, että se on ainoa näkyvä osa näkymässä. Näkymää voi lähentää, panoroida ja pyörittää. Näkymän pyörittäminen Blenderissä tapahtuu kahdella eri tapaa, Trackball- ja kääntöpöytä (Turntable) metodeilla. Trackball-metodin toiminta perustuu näkymän pyörittämiseen, kuten käyttäjä käyttäisi hiirtä, jossa on Trackball. Kääntöpöytämetodi toimii kuten levy-

soitin, jossa pyörittämiseen on käytössä kaksi akselia ja näkymä määrittää tarkemmin, mikä suunta on ylös ja mikä alas. Metodi ei ole yhtä joustava, kuin Trackball. (Blender.org 2015, viitattu 18.9.2015)

3D editorissa on eri tiloja; objekti-, muokkaus-, asento- ja kaiverrustilat, vetrex-, paino-, ja tekstuurimaalaus sekä partikkelimuokkaus. Tilaa voidaan muuttaa käyttämällä 3D editorin valikkopalkista, tai käyttäen siihen liitettyä pikanäppäintä. Tilat ovat Blenderin olio-ominaisuus, mikä tarkoittaa, että ohjelmisto on aina yhdessä määritetyssä tilassa, ja käytettävissä olevat tilat vaihtelevat valitun objektin tyypistä. Osa objekteista sallivat vain Objektitilan (Kamerat, lamput). Jokainen tila on suunniteltu tietyn aspektin muokkaamiseen objektissa. Objektit ovat valittavissa ainoastaan Objektitilassa. Muissa tiloissa objektin valinta on lukittu, lukuun ottamatta rungon Asentotilaa. (Blender.org 2015, viitattu 18.9.2015)

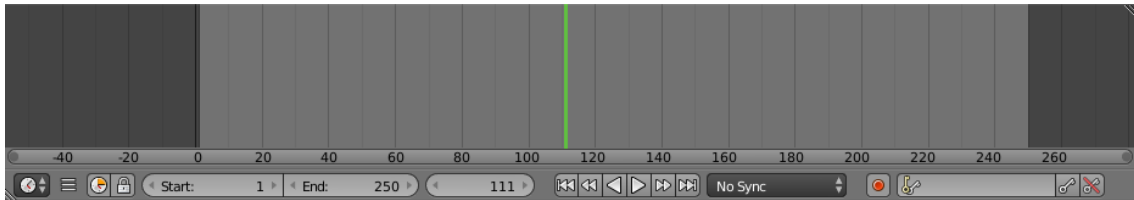
3D editorissa voidaan objektit laittaa kerroksiin (Layers). Kerroksien avulla käyttäjä voi valikoivasti näyttää objekteja tietyiltä kerroksilta, valitsemalla näytettävät kerrokset 3D näkymän valikkopalkista. Kerrokset nopeuttavat käyttöliittymän piirtokykyä, vähentää virtuaalimaailman ylimääräisyyksiä ja parantavat työnkulkua. Objektien valottumista voidaan myös hallita kerroksien avulla, säätäen valon vaikuttamaan vain valituilla kerroksilla. Renderöinnin tehokkuus myös voimistuu, kun kerroksilla voidaan määrittää, mitkä niistä renderöidään (eli mitkä objektit renderöityvät), ja mitkä kanavat/ominaisuudet ovat käytettävissä yhdistämisiin. (Blender.org 2015, viitattu 18.9.2015)

3D kerrokset eroavat 2D piirto-ohjelmien kerroksista, sillä niillä ei ole vaikutusta piirtojärjestykseen ja niiden tarkoituksena on pääosin sallia näkymän järjestely. Renderöidessä Blender renderöi vain vain valitut kerrokset. Jos kaikki valotukset ovat sijoitettuna epäaktiivisella kerroksella, renderöinnin lopputuloksessa ei näy muuta kuin objektit ympäröivällä valotuksella. (Blender.org 2015, Viitattu 18.9.2015)

4.1.2 Animaatioeditori

Blenderissä on aikajana, joka tunnustetaan kellokuvakkeesta ja on oletuksena Blenderin alareunassa. Aikajana ei itsessään ole editori, vaan informaatio- ja hallintaikkuna. Aikajanassa on käytössä kehyksiä (frame) sekä avainkehyksiä (keyframe). Kehys on kuva näkymästä tietyssä ajan-

kohdassa. Avainkehys määrittää aloitus ja lopetusajankohdan animaatiolle. Aikajanalla on hallintapainikkeet animaation pyörittämiseen, pysäyttämiseen tai kohtien ylihyppäämiseen. Aktiivisille ja valituille objekteille avainkehukset ovat esitettyinä keltaisena. (Blender.org 2015, viitattu 18.9.2015)

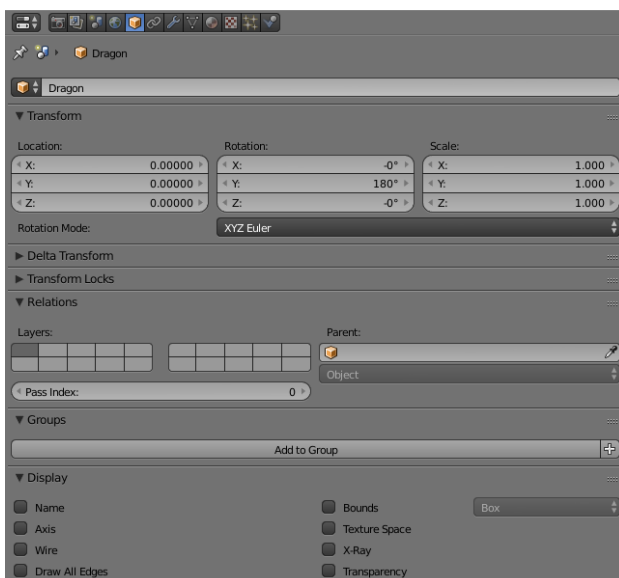


Kuvio 9. Animaatioeditori

Aikajan päänosa-alue esittää animaation kehykset aikajärjestyksessä. Sitä voi panoroida vetämällä aluetta vasemmalle tai oikealle. Oletuksena, Toisto/renderöintialue on vaalean sävyinen harmaa. (Blender.org 2015, Viitattu 18.9.2015)

4.2 Ominaisuudet -työkalurivi

Ominaisuudet -työkalurivistä löytyy suurin osa Blenderissä suoritettavista toiminnoista, joita voidaan käyttää objekteihin, materiaaleihin, animaatioon sekä renderöintiin. Ylätunnisteesta on rivi ikoneita, jotka vaihtelevat valittuna olevan objektin tyypistä. (Blender.org 2015, viitattu 18.9.2015)

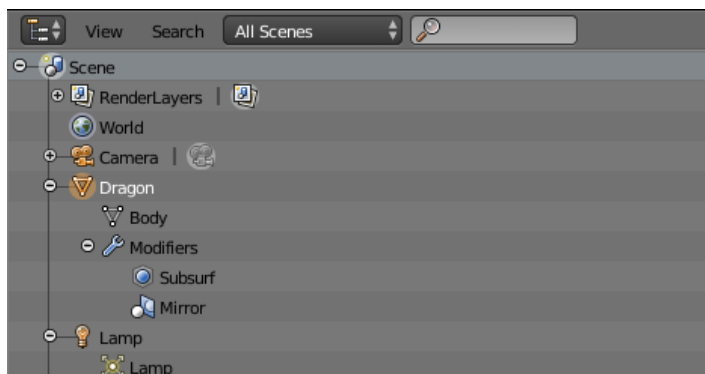


Kuvio 10. Ominaisuudet -työkalurivi

Renderöidessä työkalurivi näyttää asetukset koskien lopullisia kuvia; kuvien resoluutiota, ulostuloformaattia, suorituskykyä ja jälkikäsitteilyä koskevat. Renderointikerroksien yhteydessä työkalurivi esittää tarkemmat asetukset renderöityille kuville. Näkymien hallinnassa työkalurivi näyttää asetukset värien hallinnalle, määrittää mitä kameraa käytetään renderöintiin sekä yksikköjen ja painovoimien asetukset fyysisille mallinuksille. 3D maailman hallintaan työkalurivi tarjoaa asetukset maailman taustaväriin, sumutason, tähtien asetukset sekä asetukset ympäristön valotukselle. Itse 3D objektien hallinnalle työkalurivissä näytetään asetukset pääosaisille muotoiluille, kerroksien määrityksille sekä ryhmittelemisille. Asetukset ja mahdolliset muutokset vaikuttavat viimeisimpänä valittuun objektiin. Objektin liitoksista asetukset rajoittavat objektin liikettä animaatiotarkoituksissa. Muokkainasetuksissa voidaan asettaa objektin geometrialle muokkaimia. Objektin tietojen yhteydessä työkalurivi näyttää asetukset määriteltynä objektin tyyppin mukaisesti. Työkalurivin ikoni ja asetukset vaihtelevat valitun objektin tyyppin mukaan. Materiaaleille asetukset ovat väreille, pintojen kiiltävyydelle, läpinäkyvyydelle jne. (Wikibooks.org, viitattu 10.10.2015)

4.3 Outliner

Outliner on lista, jossa 3D objektit ovat järjestyksessä. Sen avulla voidaan nähdä näkymässä olevat objektit, poistaa ja lisätä objekteja sekä piilottaa ja näyttää ne näkymässä. Listassa voi määrittää objektin hallittavuuden, sallia sen renderöinnin ja poistaa objektien linkityksiä. Jokainen rivi listassa näyttää datalaatikon. Napsauttamalla plus-merkkiä objektien nimen vieressä, listan näkymä laajenee valittuun datalaatikkoon. Näin nähdään mitä muita laatikoita laatikko sisältää. Datalaatikon voi valita listasta, mutta valinta ei automaattisesti välity näkymään. (Blender.org 2015, Viitattu 18.9.2015)



Kuvio 11. Outliner-lista

Listan objektin kontekstivalikossa käyttäjä voi valita, poistaa valinnan, poistaa objektin ja poistaa linkin kahden elementin välillä. Dataaattikoiden näkyvyyttä voidaan hallinnoida Silmä-kuvakkeella. Objektin valittavuutta voidaan hallita hiirenosoitin-kuvakkeella. Kamera-kuvakkeella voidaan objekti laittaa näkyväksi 3D näkymässä, mutta piiloon renderöinniltä. Listan ikkunan päävalikossa on kenttä, josta voidaan kaventaa objektien listausta. (Blender.org 2015, viitattu 18.9.2015)

5 MALLINNUKSEN TOTEUTUKSEN KUVAUS

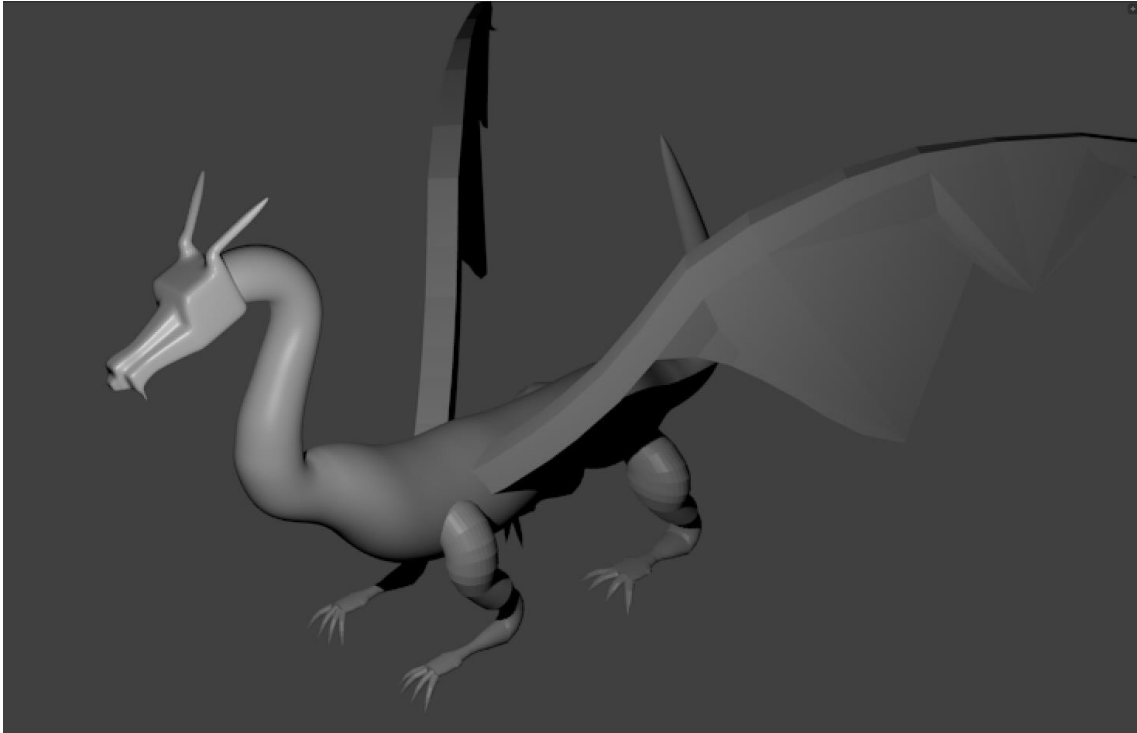
Opinnäytetyön seuraava osio alkaa luvusta 5, jossa kuvataan opinnäytteen työprosessin etene- mistä. Mallinnusprojektia lähdetään avaamaan täysin lähtöpisteistä, tarkoilla kuvauksilla mallin- nusprosessin eri vaiheista. Luku etenee mallinnuksen mekaniikan rakennukseen, jossa kuvataan mekanismin rakennuslogiikan takana vaikuttavat animaation eri osa-alueet, sekä itse mekaniikan toteutus. Luvun loppupäähän olen jättänyt objektin materiaalien ja tekstuuri- luonnin.

5.1 Rakenteen toteutus

Rakenteen toteutus kappale käsittelee mallinnuksen ulkomuodon kehitystä. Rakennetta luodessa tutkitaan eri mallinnustekniikoita ja sovelletaan yksityiskohtien merkityksellisyyttä lopputuloksen ulostulossa. Kappaleessa tuodaan esille myös mallinnuksessa kohdattuja ongelmakohtia.

5.1.1 Mallinnustekniikoiden vertailu

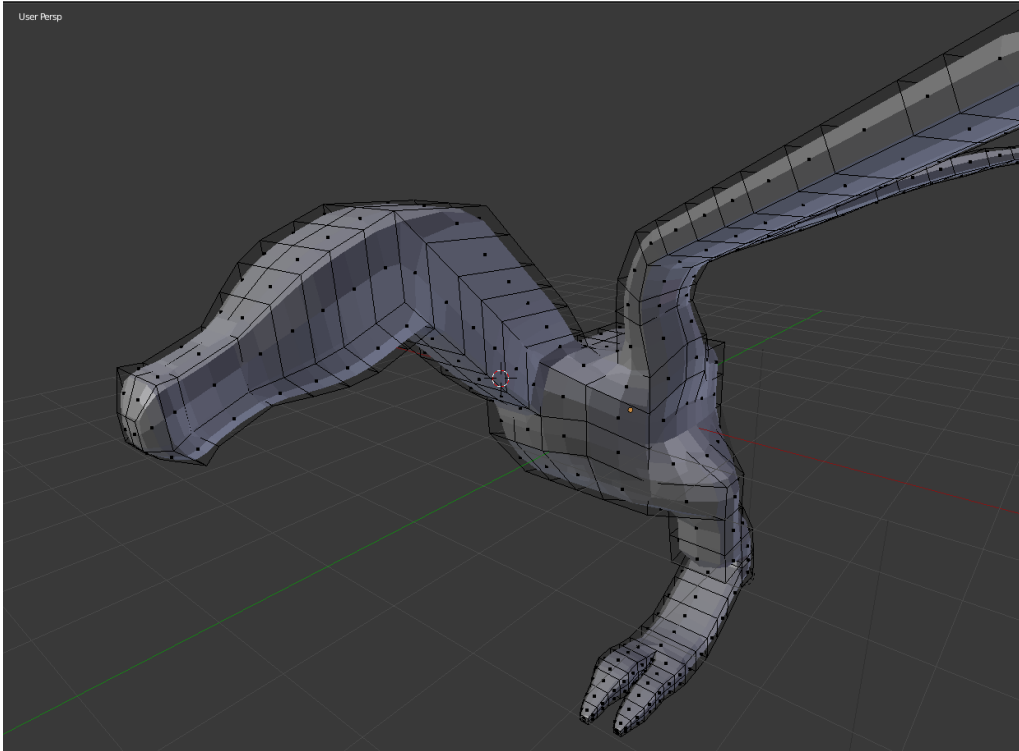
Työtä lähdettiin tarkastelemaan eri mallinnustekniikoiden (kts luku 3 Mallinnustekniikat) vertailuilla. Ensimmäistä versiota lähdettiin toteuttamaan käyttäen primitiivimallinnusta, muodostaen mallin- nuskohteen jokaisen osan valmisprimitiiveistä. Primitiivimuotojen kokoa ja muotoa käsiteltiin eri muotoiluilla, mutta lopputulos ei miellyttänyt silmää.



Kuvio 12. Primitiivimallinnettu lohikäärme.

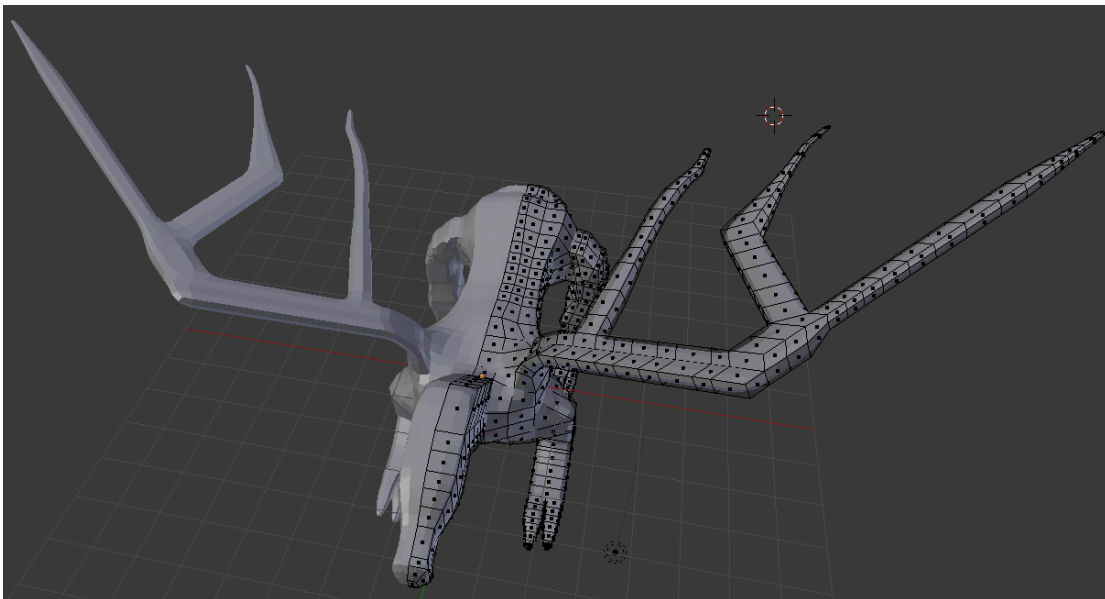
Mallinnus koostui pääosin muokatuista kuutioista ja sylintereistä. Keskivartalossa käytettiin Subdivision Surface muokkainta, jolla lisättiin polygoneja (kts luku 2.2 Objektin rakenne) ulkomuodon pehmentämiseksi. Kohteen pää on kuutio, johon on lähdetty toteuttamaan yksityiskohtia työntämällä kuution polygoneja ulospäin. Kohteen hartiat ja jalat ovat sylinterit, joiden polygoneja on työnnetty ulos ja kierretty eri akselleilla. Kynnet ovat venytettyjä, kapeaksi skaalattuja sylintereitä. Siipien alkuperä oli kuutio, jonka väliin on luotu kokeilun vuoksi yksipuolisia pintoja demonstroidaan siipien nahkaa. Tämä mallinnus jäi lopputuloksena tähän tilaan, sillä voitiin todeta sen olevan heikko toteutus.

Seuraavana lähdettiin tarkastelemaan Subdivision -mallinnustekniikkaa, sillä sen tarjoamat mahdollisuudet olivat monipuolisemmat. Mallinnustekniikka luo yhtenäisen objektin, jossa on vähemmän teräviä kulmia, ja siksi kyseinen toteutustapa valittiin pääosaiseksi mallinnukseen.



Kuvio 13. Subdivisionmallinnuksen kehikko

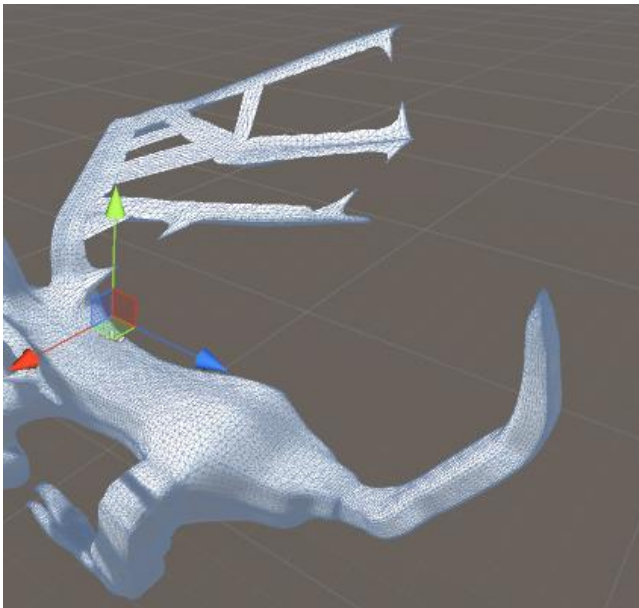
Lähtökohtana oli yksinkertainen kuutio, jonka pinnat jaettiin useammaksi polygoniksi, joita lähdettiin työntämään ulospäin. Tekniikka mahdollisti kuution reunojen pehmeiden, luoden realistisempaa kädenjälkeä. Kun mallinnuksen perusrunko oli pääosin muodostunut, otettiin käyttöön peilausmuokkain, joka kopioi luodut pinnat objektin toiselle puolelle.



Kuvio 14. Peilausmuokkaimen käyttö

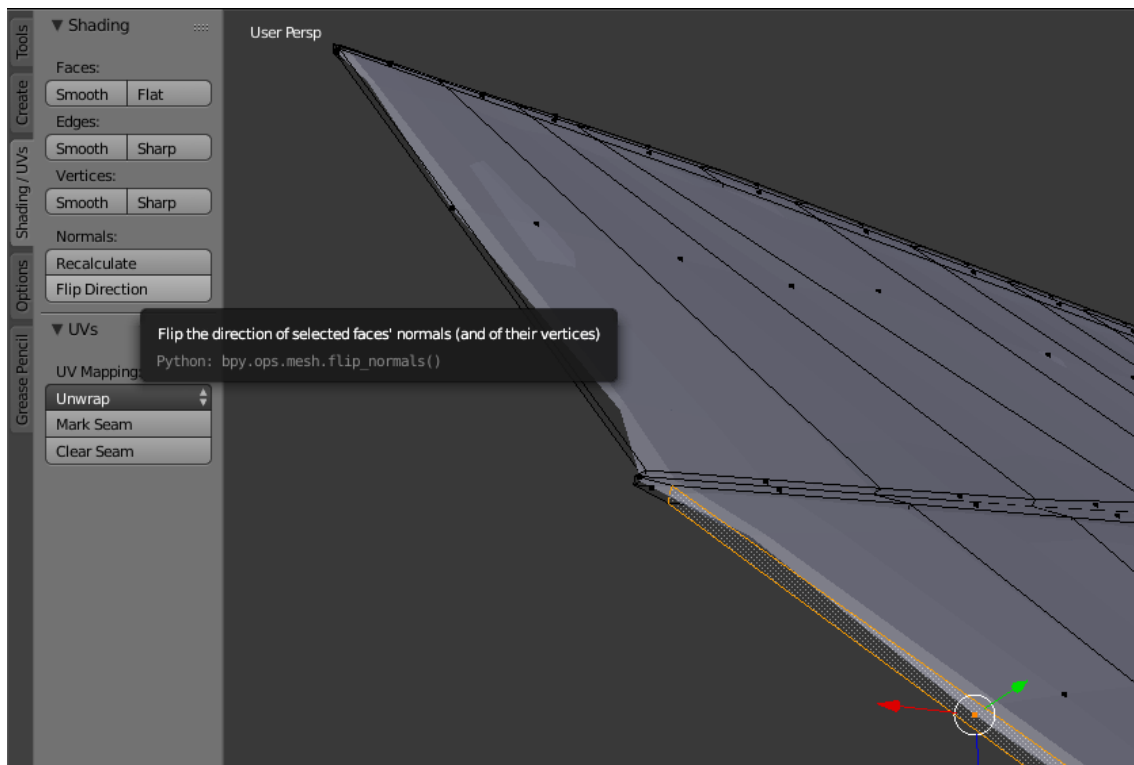
5.1.2 Siipien mallinnus

Kun mallinnus saavutti tavoiteltua muotoa, keskityttiin siipien toteutukseen. Siipiluiden välille luodut pinnat tuottivat normaalien (kts luku 2.2.2 Normaalit) vuoksi vaikeuksia, sillä Blenderin luodessa pikanäppäimellä (F) pinnan, pinta on automaattisesti yksipuolinen. Blenderissä pinnan toisen puolen näkymättömyys ei välity, mutta siirrettäessä pelimoottoriin (Unity) voitiin nähdä ongelma.



Kuvio 15. Pintojen normaalit

Ongelmaa lähdettiin tarkastelemaan kahdella tavalla; kaksipuolisten pintojen luonnilla sekä siipinahkojen rakenteen muuntamisella. Kaksipuolisten pintojen luonti osoittautui mahdolliseksi Blenderin sisäisellä toiminnolla, mutta ongelmaksi muodostui toiminnon siirto pelimoottoriin. Tästä syystä kaksipuoliset pinnat hylättiin, ja lähdettiin toteuttamaan rakenteen muutosta.

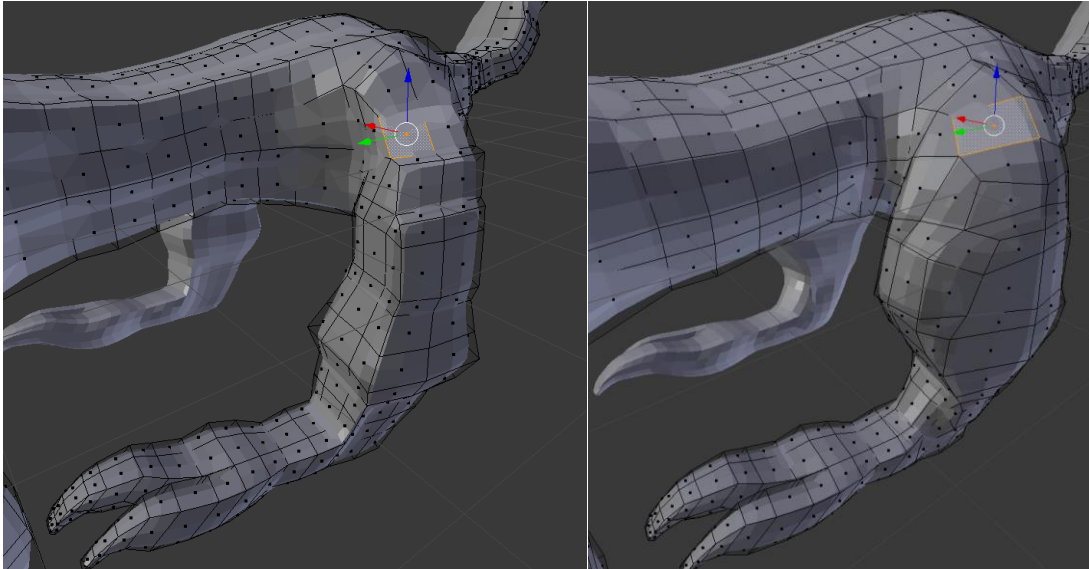


Kuvio 16. Siipien rakenteen muutos

Siipien rakenteen muutoksella tehtiin luiden molemmille puolille omat pinnat, joiden normaalit käännettiin suuntaamaan ulospäin. Näin saatiin siivet näkyviksi myös pelimoottoriin.

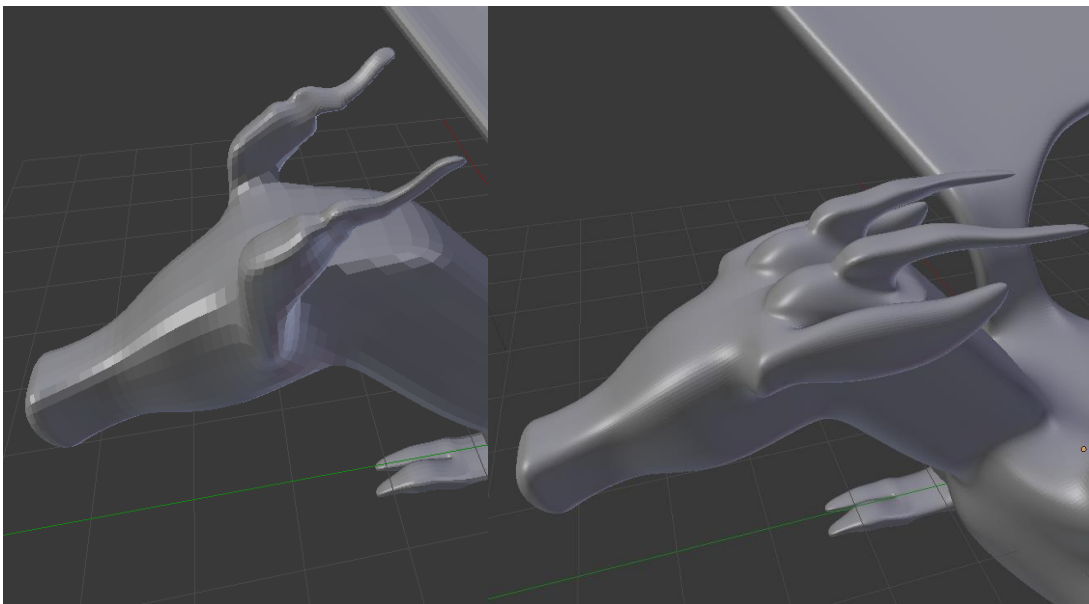
5.1.3 Mallinnuksen viimeistely

Mallinnuksen viimeistelyvaiheessa otettiin käyttöön polygon -mallinnustekniikka. Tähän mennessä työ oli mallinnettu muistuttamaan kohdetta suurpiirteisesti, mutta nyt haluttiin yksityiskohtia ja keskityttiin realismiin. Ensimmäisenä keskityttiin takajalkoihin, joita lähdettiin muotoilemaan siirtämällä verteksejä ja pintoja.



Kuvio 17. Jalkojen muotoilu

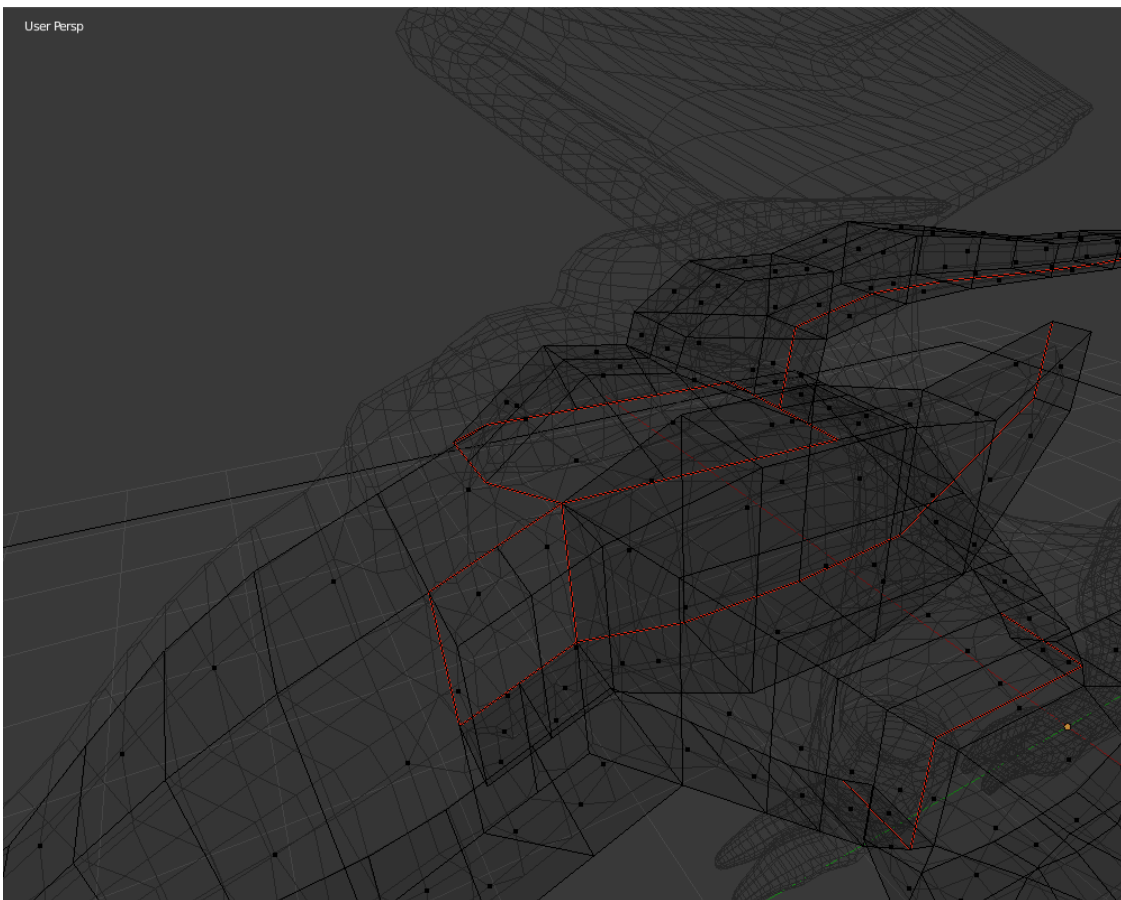
Kohteen pään yksityiskohtia tehdessä käytiin läpi muutamia vaihtoehtoja. Suurilta osin nämä olivat improvisointikokeiluja, joista sitten valittiin onnistunein toteutus.



Kuvio 18. Pään mallinnus

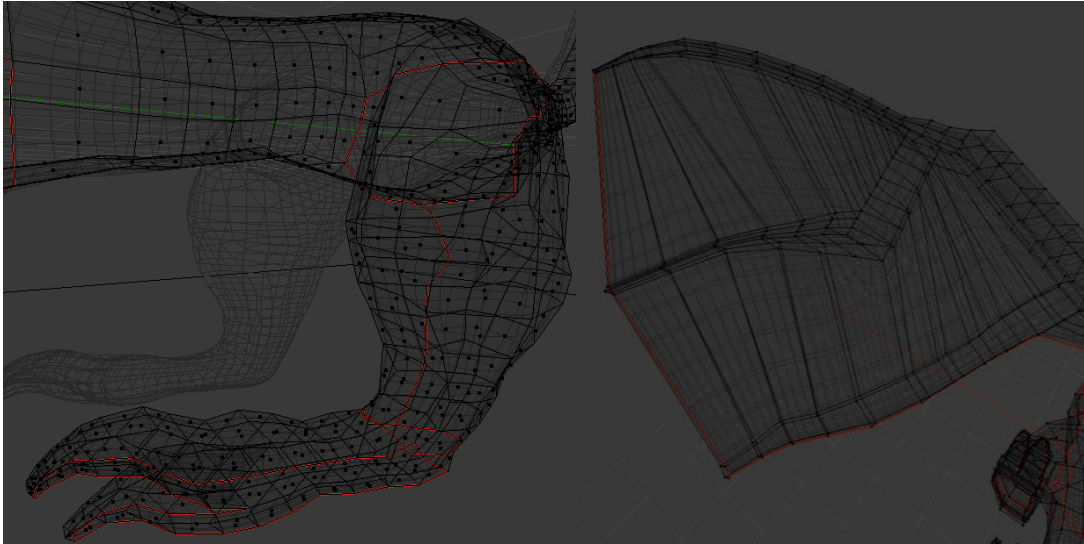
5.2 UV saumoitus

UV saumoituksen tavoite oli määrittää ääriviivat meshin osien levitykselle, kun ne sijoitettiin 2D tasolle tekstuurien maalausta varten. Saumoituksessa pyrittiin välttämään saumojen sijoittamista näkyville sijainneille, jotta teksturointivaiheessa ei tarvitsisi keskittyä saumakohtien sulauttamiseen toisiinsa. Uv-saumoitus aloitettiin mallinnuksen päästä. Pää jaettiin 3 erilliseen osaan, sarviin, korviin ja pääkokonaisuuteen.



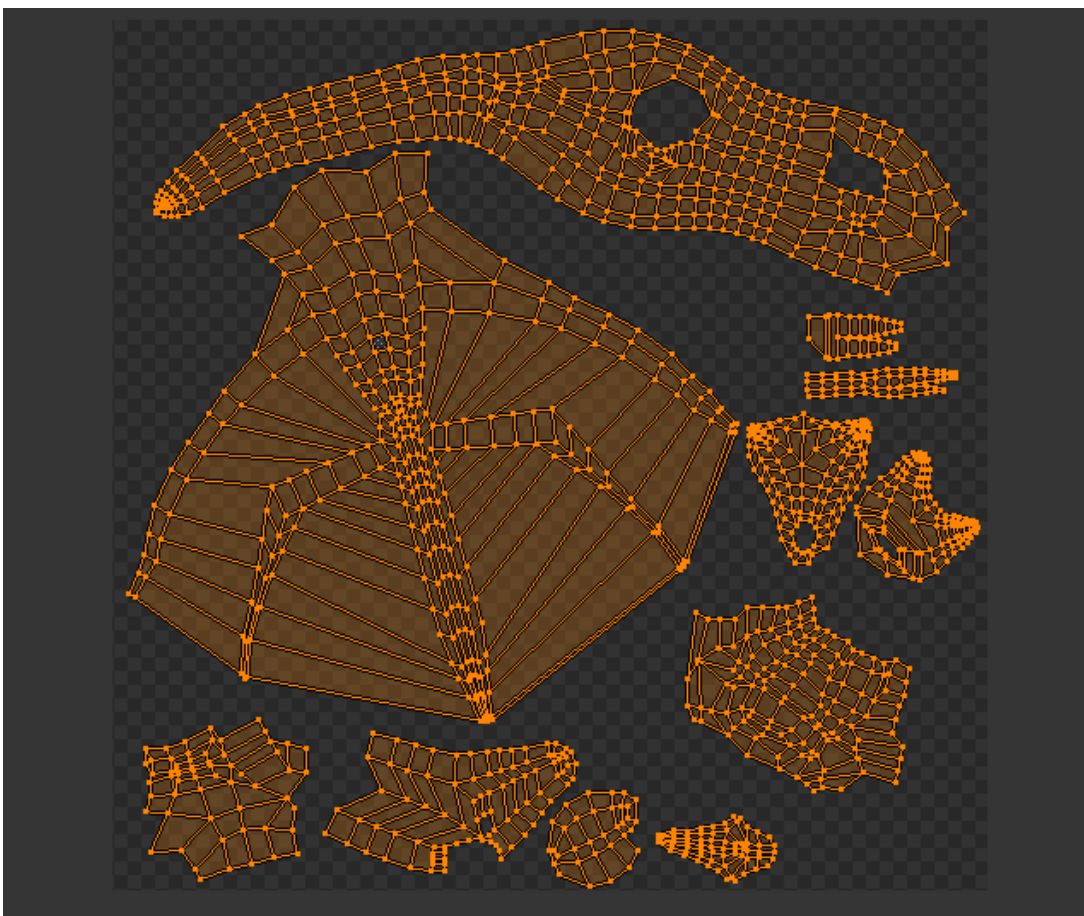
Kuvio 19. Pään UV-saumoitus

Päästä edettiin siipiin ja loppuun vartaloon. Siivet päätettiin halkaista siipien nahanpuolelta, jotta siipiluu levittyisi UV kartalle hyvin. Jalat eriteltiin reiteen, kämmeneen sekä kämmenpohjaan. Reisi halkaistiin saumalla, jotta levitys olisi selkeämpi. Sauma sijoitettiin vähiten näkyvälle sijainnille reiden sisäosaan.



Kuvio 20. jalkojen ja siipien saumoitus

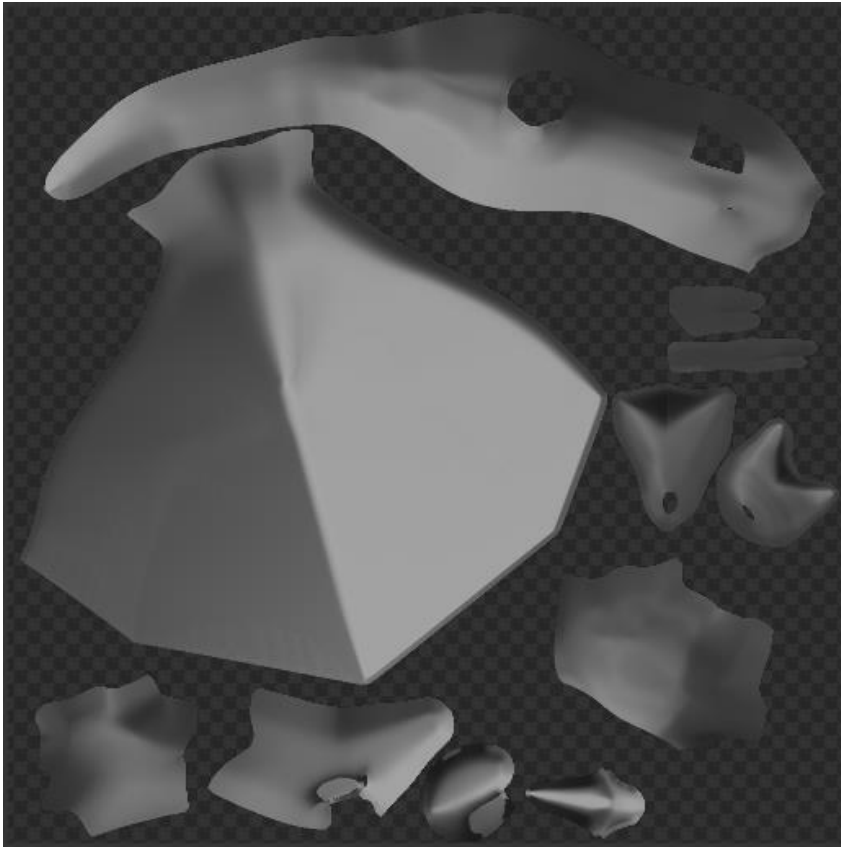
Saumoituksen jälkeen UV kartta levitettiin omaan editoriinsa. Saaret asetettiin loogiseen järjestykseen jotta tekstuurien tekeminen olisi yksinkertaisempaa. Saumoituksen johdosta saaret eivät sisältäneet päällekkäisiä verteksiryhmiä UV kartassa.



Kuvio 21. Saarien sijoittelu UV editorissa

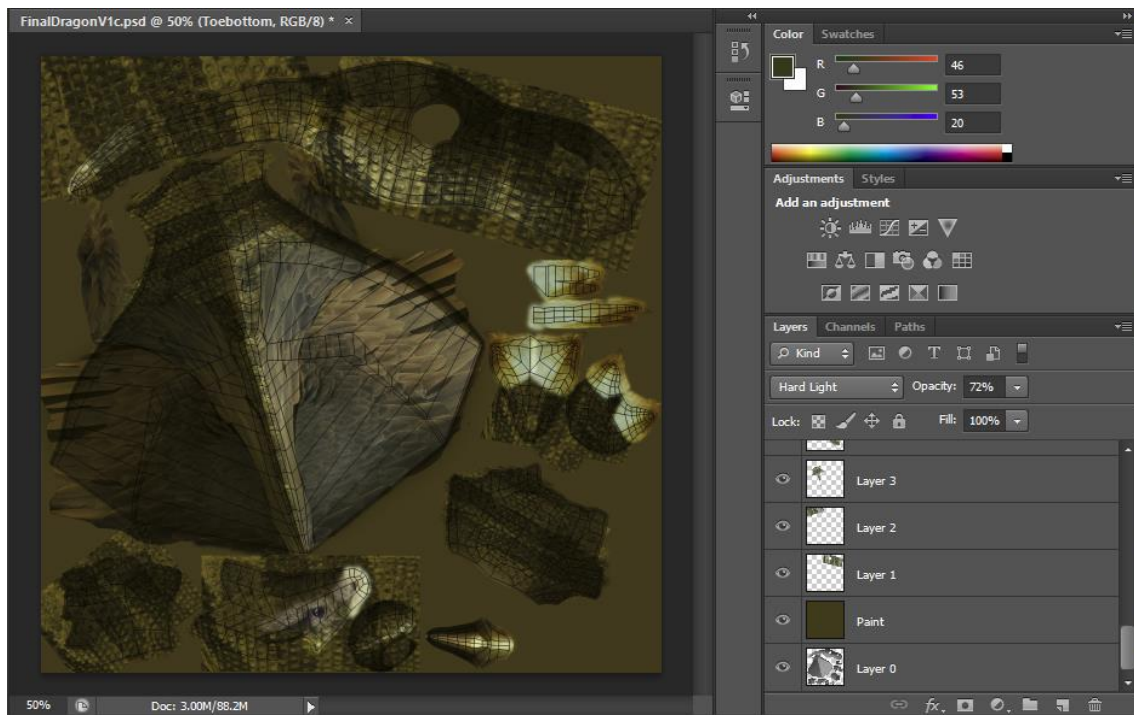
5.3 Materiaalien ja tekstuurien lisääminen

Saumoituksen valmistuttua siirryttiin UV saarien ja tekstuurikuvan käsittelyyn. Kun saaret luotiin, lisättiin UV esityksen alle uusi yksivärinen kuvatiedosto. Kuvatiedostoa ehostettiin lisäämällä Ambient Occlusion efekti (kts luku 2.6.2 Tekstuurikartat), ja se leivottiin kuvaan. Tätä vaihetta varten sijoitettiin mallinnuksen alapuolelle plane objekti, joka auttoi varjostuksen muodostamisessa.



Kuvio 22. UV Map ambient occlusion efekti.

UV saaret tallennettiin .png formaattiin, ja sama tehtiin kuvatiedostolle. Molemmat tiedostot avattiin kuva-editorissa omille tasoilleen. Tasojen väliin avattiin uusi taso tekstuurimaalaukselle, ja sen peittotyyppiä muutettiin siten, että UV editorissa luotu varjostusefekti näkyisi maalauksen lävitse. Kuvaeditorissa maalattiin teksturi hyödyntäen valmistekstuurien osia, leikattiin muista tekstuurikuvista tarvittavat osat ja liitettiin omiin tasoihin tekstuurin yhteyteen. UV saaret toimivat raja-alueina tekstuurimaalaukselle, ja niiden näkyvyys piilotettiin lopullisesta tekstuurista.



Kuvio 23. Tekstuuri kuva-editorissa.

Palattiin mallinnusohjelmistoon, ja korvattiin alkuperäinen yksivärinen kuvatiedosto juuri luodulla kuvatiedostolla. Onnistuneen saumauksen johdosta tekstuurimaalauksessa tehdyt yksityiskohdat sijoittuivat oikein meshin ympärille. Teksturointi oli nyt valmis.



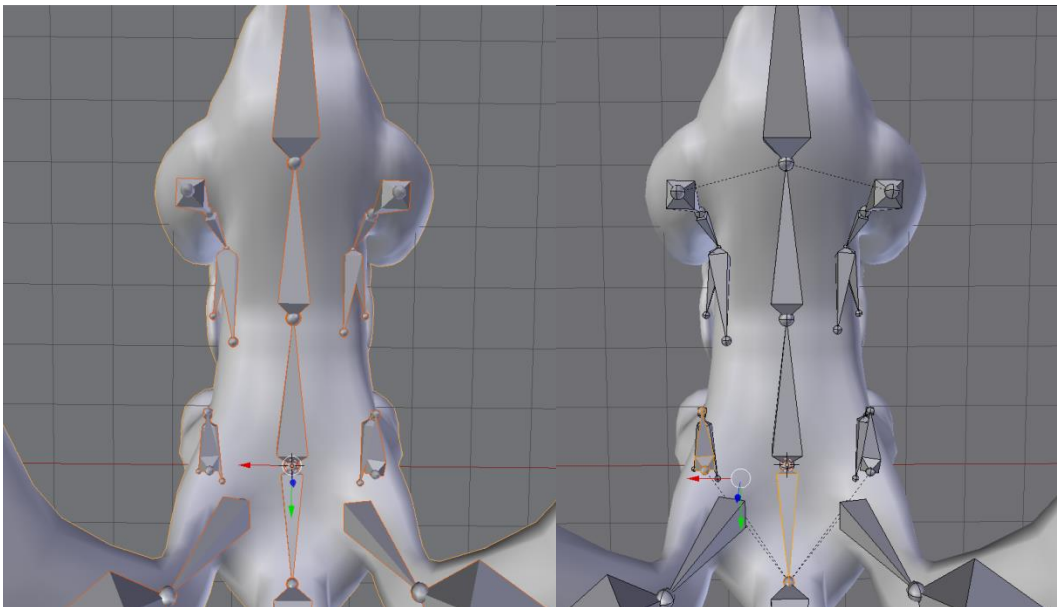
Kuvio 24. Valmis UV kartoitettu tekstuuri

5.4 Runkomekanismin luonti

Runkomekanismin periaate oli pitää se yksinkertaisena. Luuketjut rakennettiin meshin rakenteen mukaisesti, keskivartalolle, jaloille, hännälle ja siiville säädettynä, muistuttaen rakenteeltaan orgaanista luurankoa.

5.4.1 Luiden hierarkiat

Runkomekanismia lähdettiin toteuttamaan luomalla yksittäinen rungon luu, josta työnnettiin ulospäin luuketju (kts luku 2.8.1 Runko (Armature)). Luuketju sijoitettiin meshin sisälle, rakentamalla ensin rungon keskiosa päästä häntään. Keskiosasta kopioitiin jalkoihin ja siipiin omat erilliset luuketjunsä. Siipien ja jalkojen luuketjut eivät kopioinnin yhteydessä luoneet yhteyttä rungon keskiosaan, vaikka ne olivatkin kokonaisuutena Outlinerissä (kts luku 4.3 Outliner). Kun rungon keskiosan sijaintia muutettiin, jalat ja siivet jäivät paikoilleen. Tämä ratkaistiin laittamalla hierarkiaan (kts luku 2.2.4 Hierarkiat) siipi- ja jalkaluuketjujen juuriluut lapsiluiksi keskirungon rintakehän luuhun ja lantionluuhun.

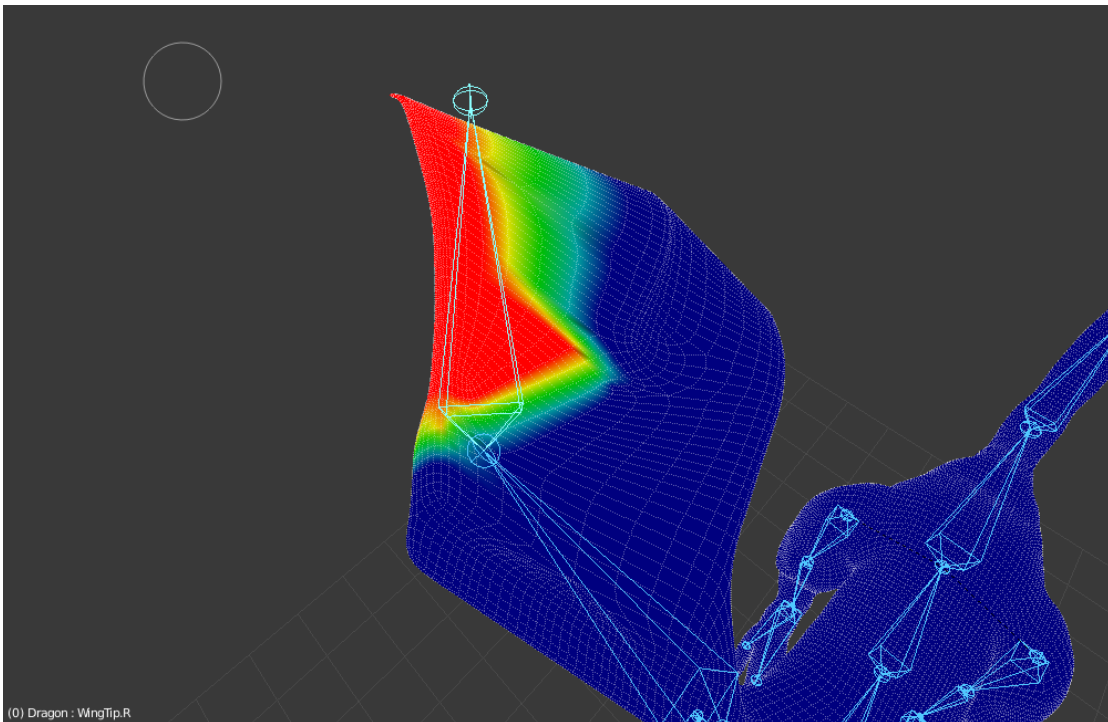


Kuvio 25. Rungon hierarkiat

Kun runko oli rakennettu kokonaisuudeksi, voitiin se laittaa hierarkiassa meshin lapsiobjektiksi. Kun meshin sijaintia muutettiin, runkomekanismi liikkui sen mukana. Voitiin siis edetä luiden muunnosarvojen vaikutusalueiden määrittämiseen, painomaalaukseen.

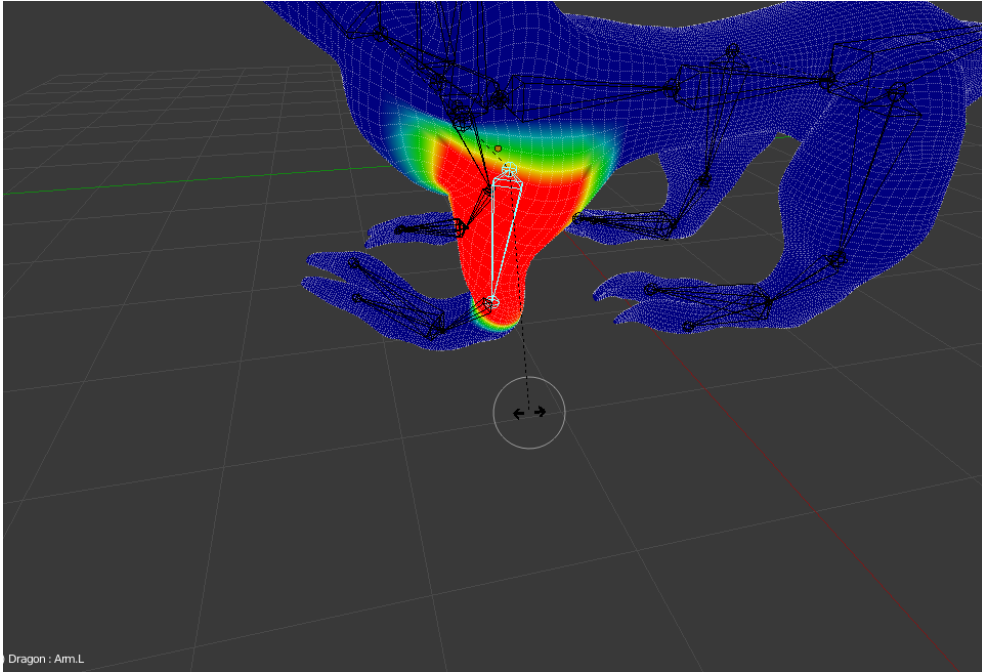
5.4.2 Painomaalaus

Työskentelytavaksi valittiin verteksiryhmien painomaalaustekniikka (kts luku 2.8.2 Meshin liittäminen (Skinning) ja asento (Posing)). Painomaalausta lähdettiin toteuttamaan haastavimmaksi koetusta alueesta, siivistä. Siipien pintojen rakenne oli soviteltu siten, että taitoskohdissa oli pintoja tiheimmin, jotta taitos ei venyttäisi tekstuureja muodottomiksi. Painomaalauksella rajattiin kärkiluun vaikutusalue koskemaan siipien rakenteen kärkialuetta.



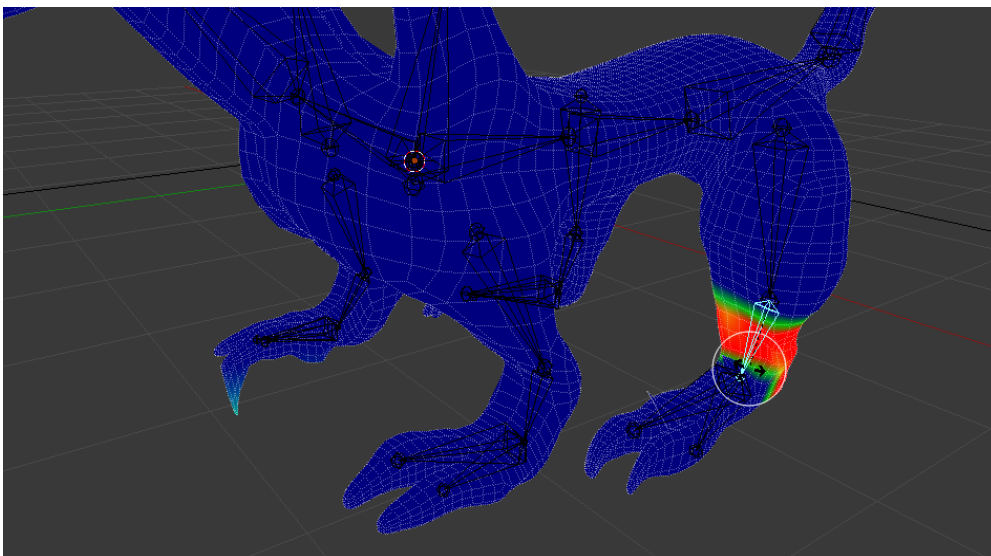
Kuvio 26. Siipien painomaalausalue

Painomaalausvaiheessa luiden hierarkiasta oli etua. Kun siiven keskiluun maalaus oli tehty, sitä taivuttaessa muutos vaikutti myös kärkiluun sijaintiin. Siipien valmistuttua voitiin edetä meshin muihin osiin. Painomaalauksessa tehtiin kiertotestejä, jotta meshin pintojen asettuminen näissä tilanteissa ei luonut rakennevirheitä.



Kuvio 27. Kiertotestit

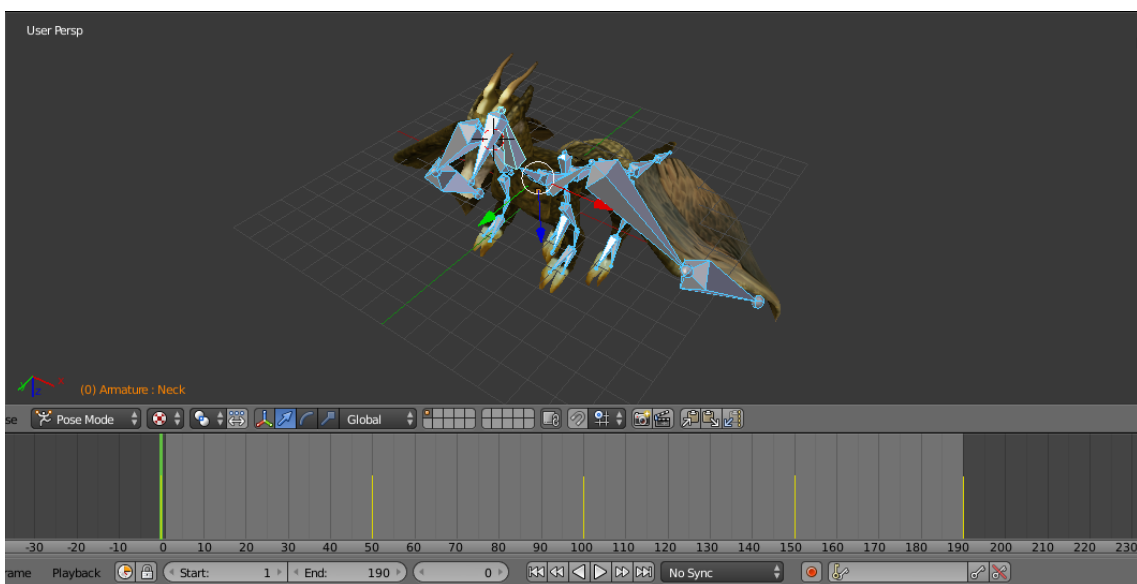
Haasteita painomaalauksessa tuotti virhemääriytykset. Riippuen käytetystä kuvakulmasta, saattoi maalattu alue yltää myös niihin osiin meshiä, jotka kuuluivat eri luiden vaikutusalueisiin. Tällöin vaihdettiin työkalua vastakkaisen tuloksen luomiseksi. Koska maalauksen painoarvo voitiin määrittää manuaalisesti, virheelliset vaikutusalueet eivät aina olleet selkeästi eroteltavissa. Tällöin virheet tuli todeta kierto- tai sijainnin muutos testeillä. Maalauksen toteuttaminen sujui meshin selkeän rakenteen vuoksi muilta osin hyvin. Nyt kaikkien luiden ollessa liitettyä omiin vaikutusalueisiinsa, voitiin siirtyä asennon ja animaation luontiin.



Kuvio 28. Painomaalauksen virheelliset vaikutusalueet

5.5 Animaation toteutus

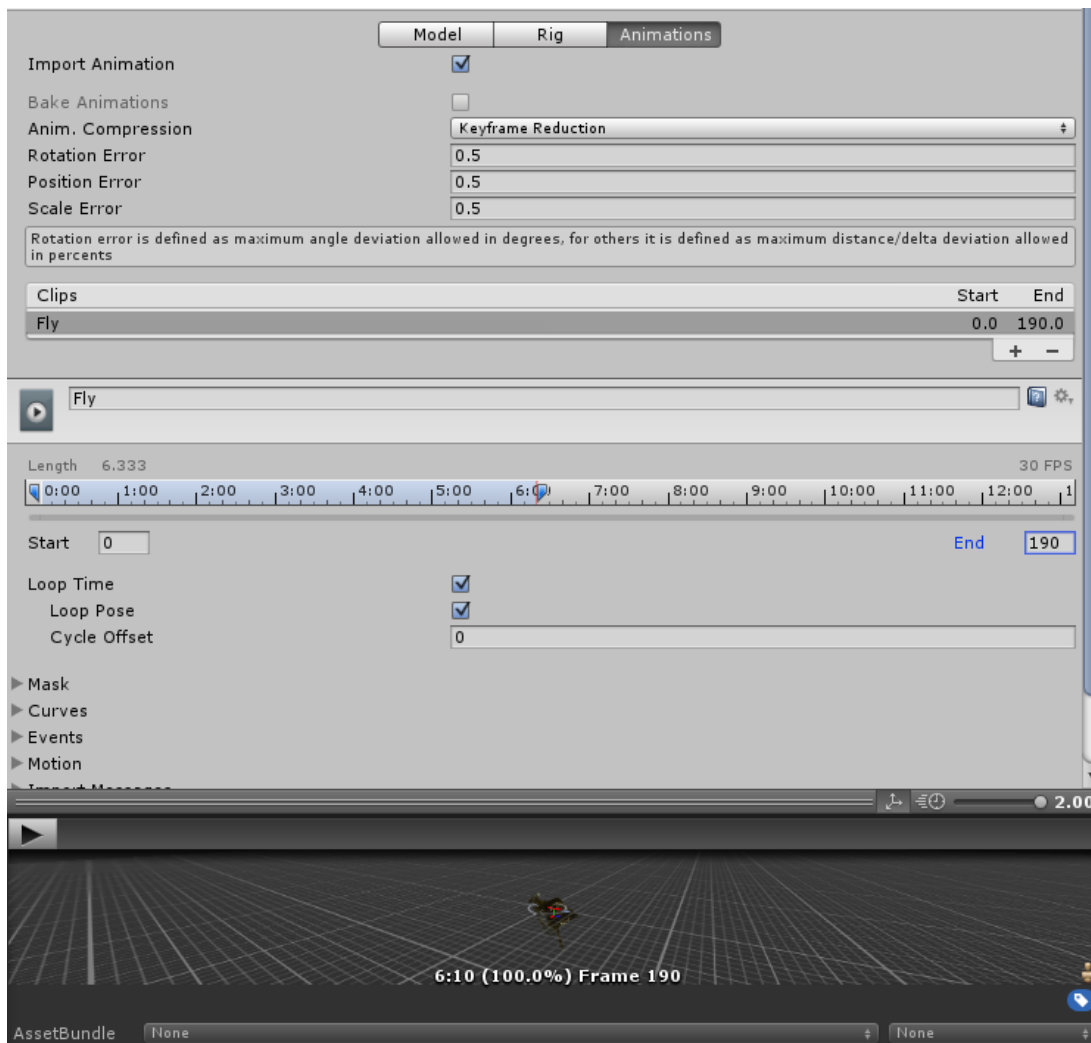
Animaatiovaiheen lähestyessä voitiin tutustua runkomekaniikan kierron toimintakuvioidiin. Oli keskityttävä meshin pintojen taivutusten luomiin rajoituksiin sekä rungon asennon eri vaiheisiin. Animaatio aloitettiin aloitusasennosta, joka tallennettiin omaksi avainkehukseksi (kts luku 2.9 Animaatio). Animaatiossa tavoiteltiin nopeatempoista lopputulosta, jonka vuoksi avainkehukset sijoitettiin tiheästi kehysten väleihin. Animaation toteutuksessa pyrittiin tähtäämään reaali maailman käyttöön. Tämä tarkoitti, että meshin liikkeet muotoiltiin muistuttamaan elävän eläimen käyttäytymistä, jolloin animaatiosta saatiin mahdollisimman uskottava.



Kuvio 29. Lentoanimaation rakennus

Jokaisessa avainkehyksessä rungon luut aseteltiin uuteen asentoon. Mekanismin rakenteen ansiosta mesh muotoutui rungon liikkeen mukaisesti eri asentoihin, luoden sulavan liikkeen kahden avainkehysten välille.

Animaation toteutuksessa oli tutkittava myös pelimoottoriympäristön yhteensopivuutta. Tästä syystä aikajanelle toteutettiin kokonaisuus, joka sisälsi useita eri animaatiosarjoja. Kokonaisuus voitiin kehysten lukuarvojen avulla leikata pelimoottoriympäristössä omiin kategorioihinsa. Tärkeintä animaatiosarjojen luonnissa oli jatkuvuus. Kun yksi sarja oli luotu, viimeinen kehys oli kopio ensimmäisestä kehuksesta.



Kuvio 30. Animaation jakaminen osiin Unity -pelimoottorissa

Kun animaationsarja oli toteutettu, lähdettiin sitä testaamaan pelimoottoriympäristössä. Animaatio jaettiin omiin osiinsa ja ne määritettiin vastaamaan eri toimintoja. Opinnäytetyö oli nyt saatettu päätökseen.

6 YHTEENVETO

Tavoitteena opinnäytteelle oli mallintaa peliprojekteihin soveltuva mallinnustyö. Tutkittiin, mitä elementtejä ja keinoja tarvitaan hyvän ja toimivan mallinnustyön toteuttamiseksi. Opinnäytteen keskittymä oli yleiskatsaus mallinnukseen ja sen tekniikoihin, jotta taitoja voidaan lähteä kehittämään ja soveltamaan useammassakin peliprojekteissa.

Työtä aloittaessani minulla oli muutamilta mallinnuskursseilta kerättyä teoriaa ja käytännöntuntemusta mallintamisesta, mutta tiedonkeruuvaiheessa käsitykseni alasta keskittyi tiiviiksi paketiksi. Taustatyö aiheenrajauksessa vei aikaa, sillä kokonaisuutena pelihahmomallinnus koostuu useamman eri alan ammattilaisen tietotaidon yhdistämisestä ja yhteistyöstä. Alan teoriaa tutkiessani sain hahmotettua, mitä mallintajan työkuvaan kuuluu, ja mitkä osa-alueet vaikuttavat mallinnuksen rakenteen luomiseen. Hyvänä esimerkkinä voidaan pitää mallintajan ja runkomekaniikan toteuttajan keskinäistä yhteistyötä, jossa ymmärrys meshin muodon taipumisesta runkomekaniikkaa soveltaessa on avain elementti oikeaoppiseen mallinnukseen.

Työn edetessä oli hyvä huomioida, että hahmon luokitus ei vaikuttanut liikaa tietoperustan keräämiseen. Opinnäytteen oli tarkoitus pysyä yleisenä katsauksena pelihahmomallinnukseen, mutta omana innoituksena toteutin hahmon fantasiaaluokassa. Oli myös tärkeää tuoda esille ohjelmistojen eroavaisuudet, esimerkiksi pintojen normaalien esitystavoissa. Vaikka mallinnusohjelmiston sisäisesti voidaan ratkaista jokin ongelma yhdellä valikon napsautuksella, ei se välttämättä kulkeudu toteutuksen mukana ja optimoidu muihin ympäristöihin. Opinnäytteen onnistumisen kannalta oli tärkeää tehdä myös testiversioita.

Blenderin valintaan vaikutti sen vapaa saatavuus, sekä se, että ohjelmisto oli jo entuudestaan tuttu harrastepohjalta. Opinnäytteen myötä ohjelmiston rakenne ja sen mahdollisuudet avautuivat uudelle tasolle, ja nyt mallintamisprosessien vaatimat ajat ja resurssit laskevat työkalun tarjoamisen eri toteutustapojen myötä. Perehtyminen opinnäytteessä esiintyviin editoreihin vaatii aikaa ja tutkimustyötä jonkin verran, vaikka perusmekaniikat olisivatkin hallinnassa. Blender tarjoaa kuitenkin kilpailevien ohjelmistojen vaatimustason mukaista sisältöä, ja on hyvä ohjelmisto aloittelevalle mallintajalle.

Opinnäytteen alkuvaiheessa ongelmia tuotti aiherajaus. Nyt kokonaisuutta tarkastellessa voidaan todeta, että taideosuuden rajaaminen työn ulkopuolelle oli oikea valinta (Konseptitaide). Kokonaisuudessaan, opinnäytteen tarjoama tietoperusta on vahva pohja taitojeni jatkokehittämiselle ja hyvä suunta urakehitykselleni.

Jatkokehityksenä opinnäytteelleni jatkan mallinnustaitojeni kehittämistä, jotta pelihahmomallinnuksien lopputulokset saisivat realistisempia ulkomuotoja ja taidokkaampia animaatioita. Tavoitteena on kehittää uraa mallintajana, ja opinnäyte on hyvä ponnistuslauta oikeaan suuntaan. Opinnäytteen ulkopuolelle jätin mallinnustyön soveltamisen pelimoottoriympäristössä, mutta demonstraation vuoksi mallinnustyöstä on tehty yksinkertainen testiversio Unity -pelimoottorilla.

Asokan, S. 2008. 3D Modeling A to Z – from concepts to techniques. Hakupäivä 20.8.2015, <http://www.sajuonline.com/Pages/Articles/technical/graphics/3d-modeling-a-z/3d-modeling-A-Z-page1.php>.

Digital-tutors. 2013. Key 3D Modeling Terminology Beginners Need to Understand. Hakupäivä 20.8.2015, <http://blog.digitaltutors.com/basic-3d-modeling-terminology/>.

deWitt, B. 2007. Basic UV mapping. Hakupäivä 9.10.2015, <http://www.modonize.com/Communities/Members/1/Bob%20deWitt/UVMMappingBasics.pdf>.

Wikibooks. 2015. Blender 3D: Noob to Pro. Hakupäivä 10.10.2015, https://en.wikibooks.org/wiki/Blender_3D:_Noob_to_Pro.

Unity. 2015. Understanding the View Frustum. Hakupäivä 12.9.2015, <http://docs.unity3d.com/Manual/UnderstandingFrustum.html>.

Unity. 2015. Materials And Shaders. Hakupäivä 11.10.2015, <http://docs.unity3d.com/Manual/Shaders.html>.

Holden, D. 2011. Subdivision Modelling. Hakupäivä 12.10.2015, <http://theorangeduck.com/page/subdivision-modelling>.

Vaughan, W. 2012. [Digital] Modelling. United States of America: Pearson Education.

Cain, J. 2011. Texturing For 3D – A Journey Through Pixels. Hakupäivä 12.10.2015, <http://motionleague.com/2011/05/texturing-for-3d-a-journey-through-pixels/>.

Blender Reference Manual. 2015. Hakupäivä 9.10.2015, <https://www.blender.org/manual/>.

Microsoft. Developer Network. What Is a View Frustum? Hakupäivä 12.9.2015,
<https://msdn.microsoft.com/en-us/library/ff634570.aspx>.

Blender Reference Manual. 2015. 3D View. Hakupäivä 9.10.2015,
https://www.blender.org/manual/editors/3dview/navigate/3d_view.html.

The Guerilla GC Project. 2008. Hakupäivä 20.8.2015,
<https://www.youtube.com/channel/UC5fpWfCQ95VFghlkH1RG70w>.

Vegdahl, N. 2014. Humane Rigging – Rigging for human beings. Hakupäivä 20.8.2015,
<https://www.youtube.com/watch?v=3e0oQyPcfql>.

King, J. 2013. Right Handed vs Left Handed Coordinate System. Hakupäivä 12.9.2015,
<https://www.youtube.com/watch?v=TGbMzoJqV7c>.

Axelsson, V. 2013. What technique is most appropriate for 3D modeling a chair for a movie production? Hakupäivä 10.10.2015,
<http://www.diva-portal.org/smash/get/diva2:638944/FULLTEXT02>.

Worldviz. 2015. Tutorial: Hierarchical Models. Hakupäivä 14.9.2015,
http://docs.worldviz.com/vizard/Hierarchical_Models.htm.