

---

# **KONTROLLIJÄRJESTELMIEN LUONTI 3D- VISUALISOINNEISSA**



Ammattikorkeakoulun opinnäytetyö

Mediatekniikan koulutusohjelma

Riihimäki, 26.11.2013

Atte Nykänen



RIIHIMÄKI  
Mediatekniikan koulutusohjelma

Työn nimi                      Kontrollijärjestelmien luonti 3D-visualisoinneissa

Nimi                              Atte Nykänen

Ohjaava opettaja              Antti Laakso

Hyväksytty                      \_\_.\_\_.20\_\_

Hyväksyjä

RIIHIMÄKI  
Mediatekniikan koulutusohjelma

---

<b>Tekijä</b>	Atte Nykänen	<b>Vuosi</b> 2013
<b>Työn nimi</b>	Kontrollijärjestelmien luonti 3D-visualisoinneissa	

---

## TIIVISTELMÄ

Tämän työn tarkoituksena on toimia ohjeistuksena rigging-tekniikoihin ja niiden soveltamiseen 3D-visualisoinneissa. Riggingillä tarkoitetaan kontrollijärjestelmien luomista 3D-mallien animoimista varten. Onnistunut rigging helpottaa animoijan työskentelyä, jolloin säästetään aikaa 3D-visualisointien animaatiovaiheessa. Riggaajan tulee pystyä soveltamaan eri tekniikoita erilaisiin malleihin ja luotujen animaatiokontrollien tulee olla tarpeeksi kattavia, jotta animaatiot voidaan toteuttaa, mutta tarpeeksi yksinkertaisia, että animoija pystyy suorittamaan tehtävänsä mahdollisimman helposti.

Työssä käydään läpi rigging-työprosessiin liittyviä tekniikoita ja 3D-sovelluksen ominaisuuksien toimintaa. Teoriaosuudessa käytettiin hyödyksi aiheeseen liittyvää kirjallisuutta ja työ toteutettiin Autodeskin 3ds Max ohjelmalla.

Työssä kerrotaan, kuinka Hämeen ammattikorkeakoulun Riihimäen yksikön mediaverstaalla toteutettiin Normet OY:n Scamec-laitteen 3D-mallin kontrollijärjestelmät 3D-visualisointia varten. Työ toimii myös ohjeistuksena mediaverstaalla tuotettaviin 3D-visualisointeihin sekä opetusmateriaalina.

**Avainsanat** 3D-visualisointi, rigging, hierarkinen linkitys

**Sivut** 29 s.

RIIHIMÄKI  
Degree Programme in Media Technology

---

<b>Author</b>	Atte Nykänen	<b>Year</b> 2013
<b>Subject of Bachelor's thesis</b>	Rigging a 3D-model	

---

ABSTRACT

The purpose of this thesis was to research rigging techniques used in 3D-visualisations. Rigging means preparing a 3D-model for animation by making a control system for the model. The purpose of the control system is to aid in the animation of model. The creation of well-made rigging is important in 3D-animation work flow as it saves time in the animation process. The rig should be versatile to allow complex animations, but the controls should be simple to understand and work with.

The thesis covers the work stages and techniques that are used in the rigging process. Related literature was used for the theoretical part of the work. The practical part of the thesis covers the rigging process of a 3D-model of Scamec-vehicle and commissioned by Normet OY. The Autodesk 3ds Max program was used in the rigging process.

This thesis also works as a guide and teaching material for rigging in 3D-visualisations in the Media Workshop at HAMK University of Applied Sciences.

**Keywords** 3D-visualisation, rigging, hierarchical linkage

**Pages** 29 p.

---

# SISÄLLYS

1	JOHDANTO.....	1
2	RIGGING .....	1
2.1	Hierarkiat.....	3
2.2	Linkitys.....	4
2.3	Kinematiikka .....	4
2.3.1	Forward kinematics .....	5
2.3.2	Inverse kinematics .....	5
2.4	Constraints.....	5
2.5	Helpers .....	7
2.5.1	Dummy helper .....	7
2.5.2	Point Helper.....	7
2.5.3	Tape Helper .....	8
2.6	Wire Parameters .....	9
2.6.1	Parameter Wiring Dialog.....	10
2.7	Parameter Editor.....	10
2.8	Attribute Holder .....	11
2.9	Parameter Collector.....	12
2.10	Schematic View.....	12
3	SCAMEC-MALLIN RIGGINGPROSESSIN TYÖNKULKU.....	13
3.1	CAD-mallin valmistelu .....	13
3.2	Riggaaminen.....	18
3.2.1	Linkittäminen .....	18
3.2.2	Ohjaimet ja constraintit .....	19
3.2.3	Renkaiden pyöriminen.....	23
3.2.4	Animaatiokontrollien asettaminen.....	25
3.3	Valmis malli .....	27
4	YHTEENVETO .....	28
	LÄHTEET .....	29

## TERMISTÖ

Child	Objekti, joka on linkitetty toiseen objektiin.
Constraint	Rajoite, jonka avulla määritetään objektien liikkeitä 3D-animaatiossa.
Dummy	Tarkoituksellisesti toimimaton laite tai työkalu, jota käytetään toiminallisen laitteen tai työkalun tilalla.
End Effector	Hierarkisen rakenteen kontrollipiste, loppuvaikuttaja. Loppuvaikuttajan liikuttaminen vaikuttaa ylöspäin hierarkiassa kaikkiin ketjun osiin, muokaten niidenkin suuntausta ja paikkaa loppuvaikuttajan liikkeen perusteella.
Helpers	Apuobjektit, jotka auttavat animaatioliikkeiden luomisessa sekä monimutkaisten hierarkioiden rakentamisessa.
Inverse/Forward Kinematics (IK/FK)	Määrittävät hierarkisen ketjun kontrollipisteiden toiminnan.
IK ketju	Hierarkkinen animaatioketju, jossa liikutaan alhaalta ylöspäin.
IK solver	Laskee IK ketjun osien muutokset animoidessa.
Keying	Avaintaminen, avainkehysten asettaminen aikajanalle.
Keyframe	Animaattorin aikajanalle asettama avainkehys, joka sisältää valittujen kappaleiden paikkatietoja. Avainkehysten välillä tapahtuva muutos muuttuu animoiduksi liikkeeksi.
Modifier	Objektiin asetettava muuttuja, jolla muokataan objektin parametrejä.
Parent	Objekti, johon on linkitetty yksi tai useampi objekti.
Pivot	Objektin paikallisen koordinaatiston keskipiste. Objektin kääntyminen ja koon skaalautuminen tapahtuu tämän pisteen ympärillä. Oletusarvoisesti kappaleen keskellä.
Rig	Järjestelmä, jonka avulla animoija pystyy hallitsemaan ja kontrolloimaan animoitavaa objektiä tietyllä tavalla.
Rigging	Riggingissä luodaan kontrollijärjestelmä, jota animoija käyttää animaatioprosessissa.
Scene	Virtuaalinen ympäristö, jolla 3D objektit sijaitsevat. Vapaasti suomennettua termiä skene on käytetty tässä opinnäytetyössä.
Viewport	3D-ohjelman ikkuna, jossa 3D-mallien käsittely tapahtuu

## 1 JOHDANTO

3D-visualisointien työnkulku jakautuu useaan osaan, kuva 1. Ensin mallintaja luo 3D-mallin ja lisää siihen tarvittavat materiaalit. Ennen kuin voidaan siirtyä animaatiovaiheeseen, luodaan kontrollit animoijaa varten. Animaatiokontrollien luomista 3D-visualisointeja varten kutsutaan riggingiksi. Kun animaatiot on toteutettu, asetetaan valaistus sekä kamerat, jotka määrittävät miltä animaatiot näyttävät lopullisessa videossa. Viimeinen vaihe on renderöinti, jossa lopullinen animaatiovideo luodaan. (Autodesk 3ds Max Help – Project Workflow). Tämä opinnäytetyö käsittelee rigging-vaihetta ja sen tärkeyttä 3D-visualisointiprojektien onnistumiselle.



Kuva 1. 3D-visualisointien työnkulku

Koska 3D-mallit rakentuvat useista objekteista, on objektien animoiminen yhdessä lähes mahdotonta 3D-tilassa, jos objektien liikkeet eivät vaikuta toisiin objekteihin, koska jokaista objektia jouduttaisiin animoimaan erikseen, jotta päästään haluttuun lopputilanteeseen. Tätä varten on luotu helpompi tapa liikutella malleja ja tätä tapaa kutsutaan riggingiksi. (Vasconcelos, 5.)

3D-animaatiossa rigging ja itse animaatio ovat lähtökohtaisesti eri osat työprosessissa, mutta ne linkittyvät oleellisesti toisiinsa. Ilman riggingiä animaation toteuttaminen on hankalaa, ja pelkän rigin rakentamisella ei saada animaatioita tehtyä. (Beane, 6.1.)

Tässä opinnäytetyössä on tarkoituksena perehtyä riggingin työprosessiin, käytettäviin työkaluihin ja niiden soveltamiseen, kun luodaan 3D-visualisointeja. Hämeen ammattikorkeakoulun (HAMK) Riihimäen yksikön mediaverstaalla tuotetaan 3D-visualisointeja asiakkaille, joten opinnäytetyö toimii pohjustuksena sille, kuinka työprosessia voidaan kehittää parempien lopputuloksien aikaansaamiseksi. Opinnäytetyö toimii myös ohjeistuksena tekijöille.

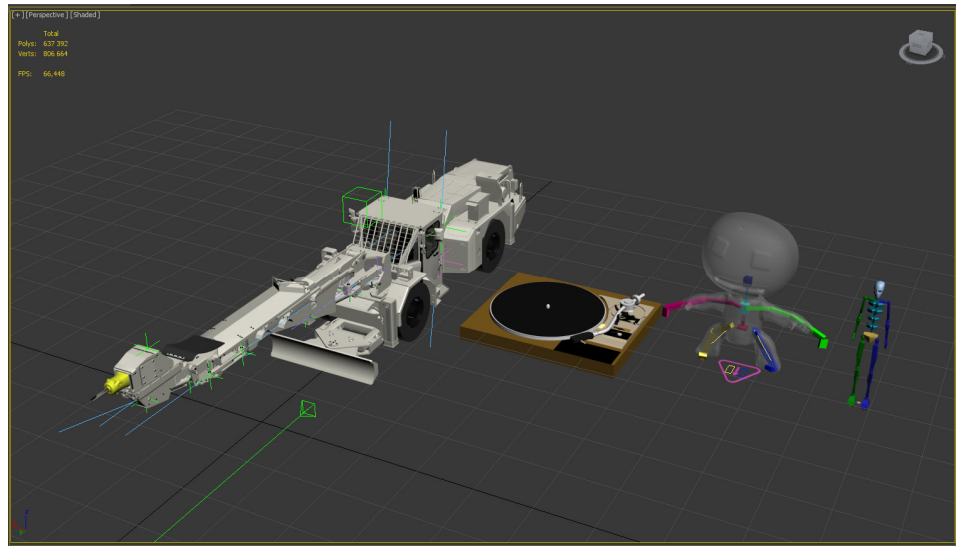
Tässä työssä on käytetty Autodeskin 3ds Max-ohjelmaa ja sen ominaisuuksia. Samankaltaisia ominaisuuksia löytyy myös muista 3D-mallinnusohjelmista, koska 3D-animaation määritelmät ja termit eivät ole ohjelmasidonnaisia.

## 2 RIGGING

3D-animaatiossa tarvitaan jonkinlaista järjestelmää, jonka avulla animoija pystyy hallitsemaan ja kontrolloimaan animoitavaa objektia tietyllä tavalla. Tällaista kontrollijärjestelmää kutsutaan rigiksi (Beane, 6.1.)

Riggingissä luodaan kontrollijärjestelmä, jota animoija käyttää animaatioprosessissa. Riggaaja ja animoija voivat olla eri henkilö. Riggaajan tehtävänä on helpottaa animoijan työtä siten, että animoija näkee käyttöliittymässä vain sen mitä tarvitaan objektien animoimiseen. (Beane, 6.1.)

Rigien yleisin käyttö on hahmoanimaatioissa, mutta samaa tekniikkaa voidaan käyttää minkä tahansa 3D-mallin valmistelussa animaatiota varten, kuva 2. (Wikipedia 2013a.)



Kuva 2. Erilaisia rigejä.

Riggingiä kutsutaan usein yhdeksi vaikeimmista aiheista 3D-animaatiossa. Luotaessa rigiä on kaksi tärkeää ohjenuoraa: rigin tarvitsee olla tarpeeksi yksinkertainen animoijalle ja rigin pitää olla tarpeeksi kattava, jotta saadaan luotua uskottavia liikkeitä 3D-mallille. Jos rig on liian yksinkertainen, animoijalle saattaa olla hankalaa tuottaa halutunlaisia animaatioita, mutta samalla liian hankala rig saattaa hidastaa animoijan työskentelyä. (Vasconcelos, 21.)

Rigin luomiselle ei ole yhtä oikeaa tapaa. Jokainen rig rakentuu omanlaisesta järjestelmien ja kontrollien hierarkiasta, jotka yhdessä toimivat tietyssä järjestyksessä 3D-mallin animoimiseksi. Tämä hierarkia on perusmuodossaan parent/child-suhde, jossa yksi objekti on parent ja toinen objekti child. Child-objekti pystyy liikkumaan, kääntymään ja skaalautumaan parent-objektista riippumatta, mutta kun parent-objekti liikkuu, child-objekti seuraa perässä. Yhdellä parentilla voi olla monta childia ja myös childillä voi olla omia childeja. (Beane, 6.1.)

Luotaessa rigiä työprosessi etenee osissa. Ensin luodaan malli, joka voi koostua useista osista. Mallin osat linkitetään toisiinsa hierarkkisesti ja määritetään pivot-pisteet, eli osien paikallisen koordinaatiston keskipisteet, joiden ympäri osa kääntyy. Tämän jälkeen lisätään rakennettuun hierarkiaan IK solverit, joka määrittää osien liikkeet. IK-solverien tehtävänä on laskea, kuinka objektien liikkeet vaikuttavat muihin



objekteihin, kun pyritään saavuttamaan tietty loppupiste. (Autodesk 3ds Max Help – IK Solvers). Yksinkertaisissa 3D-animaatioissa ohjelma luo itse käänteiset kinemaattiset animaatiot ilman, että IK-solveereita tarvitsee itse asettaa. Yleensä joudutaan luomaan useita IK ketjuja hierarkiaan ja usein halutaan luoda myös useita eri hierarkioita, jotka eivät ole yhteydessä toisiinsa animaation helpottamiseksi. Tämän jälkeen määritellään, kuinka osat käyttäytyvät pivot-pisteissä, esimerkiksi asettelemalla rajoja tai haluttuja kulmia objekteihin käyttämällä constrainteja. (Autodesk 3ds Max Help – Inverse Kinematics.)

Kun rig on rakennettu, animoidaan joko liike loppupisteeseen tai end effector, eli ketjun viimeisen objektin liike. Tällöin animaatio vaikuttaa kaikkiin IK ketjun komponentteihin. Objektien kontrolloimista varten voidaan lisätä constrainteja, jotka auttavat hallitsemaan animaatiota. (Autodesk 3ds Max Help – Inverse Kinematics.)

Animaatiot toteutetaan käyttämällä animaatiokontrolleja, jotka luodaan rigginprosessissa. Animaatiot tehdään keyframe-animaatiolla, eli animoija luo aikajanalle avainkehysiksi, jotka sisältävät valittujen kappaleiden paikkatietoja. Avainkehysten välillä tapahtuva muutos muuttuu animaatioliikkeeksi, kun tietokone laskee kappaleiden sijainnit kehysten välillä. Kehysten asettamista kutsutaan avaintamiseksi (keying). (Beane, 6.21.)

### 2.1 Hierarkiat

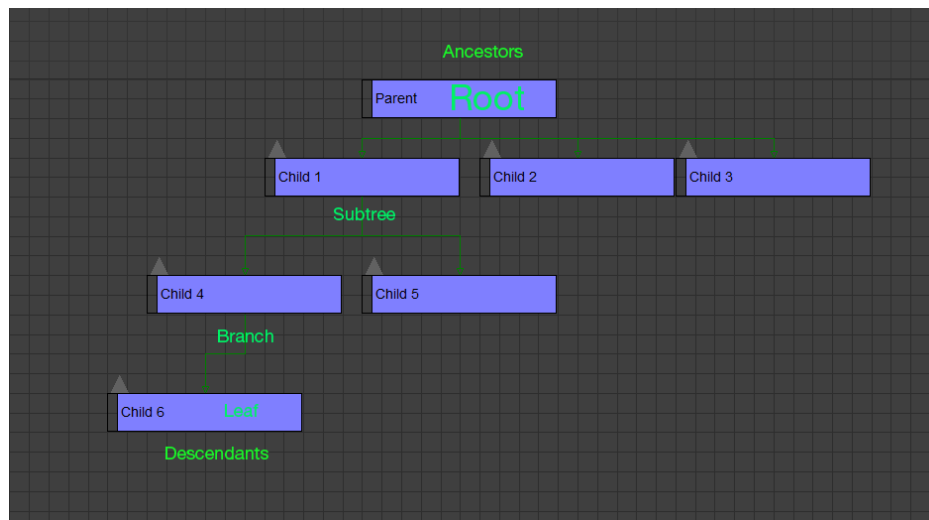
Kun luodaan 3D-animaatioita, yksi tärkeimmistä ominaisuuksista on kyky linkittää objekteja yhteen, jolloin muodostuu ketju. Linkittäessä objekteja toiseen, syntyy parent/child-suhde. Muutokset parent-objektiin siirtyvät child-objekteihin. Tätä ketjua kutsutaan hierarkiaksi.

Hierarkioita käytetään linkittämään useita objekteja yhteen parent-objektiin, jotta niitä voidaan helposti animoida ja muokata siirtämällä, kääntämällä tai skaalaamalla parenttia. Hierarkiaa käytetään myös linkittämään kamera tai valo toiseen objektiin, jotta se seuraa objektia läpi skenen. Linkitetyillä objekteilla voidaan simuloida nivel-rakenteita, jotta voidaan animoida hahmoja tai mekaanisia laitteita. Kun linkitetään objekteja dummy-objekteihin, voidaan luoda monimutkaisia liikkeitä yhdistämällä useita yksinkertaisia liikkeitä.

Hierarkiassa linkitettyjen objektien suhde on verrattavissa sukupuuhun. Parent on objekti, joka kontrolloi yhtä tai useampaa lasta. Parent-objektia ohjaa usein toinen ylempi parent-objekti. Child-objektia ohjaa sen parent ja child voi olla myös parent muille childeille. Esivanhemmat ovat kaikki child-objektin parentit ja esivanhemmat ja jälkipolvella viitataan kaikkiin childeihin ja niiden lapsenlapsiin.

Hierarkia on kaikki parentit ja childit linkitettyinä yhteen rakennelmaan, kuva 3. Root, eli juuri, on hierarkiassa kaikkein ylin parent. Subtreellä viitataan tietyn parentin jälkeläisiin. Branch, eli oksa, on polku yhdestä

parentista tiettyyn jälkeläiseen. Lehti (leaf) on child-objekti, jolla ei ole yhtään lasta. (Autodesk 3ds Max Help – Hierarchies.)



Kuva 3. Hierarkian osat.

## 2.2 Linkitys

Ennen kuin riggaaminen aloitetaan, on tärkeää suunnitella ensin, kuinka osien linkitys toteutetaan. Linkittämisellä ja sen toteutuksella on suuri vaikutus mallin käytettävyydelle: loogisesti toteutettu linkitys mahdollistaa monimutkaisten animaatioiden toteutuksen, mutta jos linkitys ei vastaa tarvetta, animaatioiden luominen voi olla hankalaa tai jopa mahdotonta. Objektien hierarkkiselle linkitykselle tehty strategia voidaan jakaa kahteen periaatteeseen: linkityksen tulee edetä loogisesti parentista childiin ja parent-objektit liikkuvat vähemmän kuin sen jälkeläiset.

Kun edetään parentista childiin, linkitysten ei tule hyppiä objektista toiseen. Jos kaksi objektia koskettaa toisiaan, tulisi niiden todennäköisesti olla linkitettyinä parentiksi ja childiksi. Linkittämistä pystyy helpottamaan käyttämällä useita hierarkioita yhden sijaan. (Autodesk 3ds Max Help – Linking Strategy.)

## 2.3 Kinematiikka

Kinematiikalla tarkoitetaan geometrista liikeoppia, joka tutkii kappaleiden liikettä geometriseltä kannalta, kiinnittämättä huomiota liikkeen syihin. Liikkeen kuvaamiseksi, kinematiikka tutkii pisteiden, linjojen ja muiden geometrinen objektien liikkeitä sekä niiden ominaisuuksia kuten nopeutta ja kiihtyvyyttä. Kinematiikkaa käytetään kuvaamaan useiden toisiinsa yhdistyneiden osien, jotka on linkitetty toisiinsa, liikkeitä. Esimerkkejä tällaisista systeemeistä ovat robottikädet, moottorit sekä ihmisluurangot. (Wikipedia 2013b.)

3D-animaatiossa kinematiikalla kuvataan ketjutettua animaatiota tai liikettä ja kinematiikalle on kaksi tyyppiä: Forward Kinematics (FK), jossa parent-objekti liikuttaa sen jälkeen tulevia objekteja sekä Inverse Kinematics, jossa child-objektin muutokset vaikuttavat sitä ylempänä ketjussa oleviin objekteihin. Forward Kinematics on yksinkertaisin tapa animoida hierkioita, mutta Inverse Kinematics antaa mahdollisuuden tuottaa monimutkaisia animaatioita, kuten hahmoanimaatioita ja monimutkaisia mekaanisia animaatioita, helpommin kuin FK-animaatiot. (Autodesk 3ds Max Help – Hierarchies and Kinematics.)

### 2.3.1 Forward kinematics

Forward Kinematicsissa (FK), eli suoraan etenevässä kinematiikassa, edetään hierarkkisesti parentista childiin. Pivot-pisteiden asettelu määrittää yhdistävän liitoksen. Objektin sijainti, rotaatio ja asento periytyvät parentista childiin. Esimerkiksi ihmisruumiin hierarkkisessa linkityksessä kehon (parent) taittuessa pää (child) seuraa mukana, mutta päätä voi kääntää ilman että se vaikuttaa kehoon. (Autodesk 3ds Max Help – Forward Kinematics.)

### 2.3.2 Inverse kinematics

Inverse kinematics (IK), eli käänteinen kinematiikka, on animaatiotapa ja asettelutapa, joka rakentuu hierarkkisen linkityksen konseptin päälle. Jotta pystyy ymmärtämään, kuinka IK toimii, pitää ensin tietää hierarkkisen linkityksen ja Forward Kinematicisin periaatteet. (Autodesk 3ds Max Help – Inverse Kinematics.)

Inverse kinematics kääntää ketjun manipulaation suunnan. Child-objektien liike vaikuttaa myös parent-objektien asentoon. Esimerkiksi ihmisen ranteen liike vaikuttaa muun käden asentoon. (Autodesk 3ds Max Help – Inverse Kinematics.)

3D-animaatiossa käänteisellä kinematiikalla pystytään nopeasti luomaan ja animoimaan monimutkaisia liikkeitä, koska IK solverit laskevat avainkehysten väliset muutokset objektien sijainnissa ja asennossa ilman, että animoija joutuu itse vaikuttamaan objekteihin. (Autodesk 3ds Max Help – IK solver.)

## 2.4 Constraints

Constraintit ovat erityisiä ohjaimia joiden avulla animaatioprosessia pystytään automatisoimaan. Constrainteilla pystytään ohjaamaan objektien asentoa, rotaatiota tai skaalaa määrittämällä niiden suhteita toisiin objekteihin. Constraintit tarvitsevat animoidun objektin sekä vähintään toisen objektin jota käytetään kohteena. Kohde asettaa tietyt animaatorajoitteet rajoitettuun objektiin. Esimerkiksi objekti pystytään asettamaan liikkumaan tiettyä polkua pitkin käyttämällä path constraintia

ja keyframe animaatiolla voidaan määrittää millä aikavälillä rajoite vaikuttaa objektiin. (Autodesk 3ds Max Help – Animation Constraints.)

Constaranteilla on useita eri käyttötapoja. Constrainteja voidaan käyttää linkittämään objekti toiseen tietyllä aika välillä, linkittämään objektin liike tai rotaatio yhteen tai useampaan objektiin ja pitämään objektin sijainti kahden tai useamman objektin välissä. Objekteja voidaan myös rajoittaa seuraamaan polkua tai pysymään pinnan päällä. Constrainteilla saadaan myös objektit katsomaan toiseen objektiin tai pitämään objektin relaatiota toiseen objektiin samanlaisena. (Autodesk 3ds Max Help – Animation Constraints.)

Autodeskin 3DS Max sisältää useita Constraint-ohjaimia, joiden avulla animaatioita pystytään automatisoimaan. (Autodesk 3ds Max Help – Animation Constraints.)

Attachement Constaraintilla rajoitetaan objektien sijaintia siten, että objektin sijainti on liitetty toisen objektin faceen eli tiettyyn osaan objektin pinnasta. Attachment parametreista valitaan objekti johon objekti sidotaan, voidaan asettaa avainnukset (keying) objektin sijainnille sekä muuttaa TCB-kontrollien (jännitys, jatkuvuus, kallistus) arvot. (Autodesk 3ds Max Help – Attachment Constraint.)

Link Constraint linkittää objektin sijainnin, rotaation ja koon toisen objektin sijaintiin, rotaatioon ja kokoon. Linkitys on oleellinen osa 3D-mallien riggausta, koska linkit määrittävät miten mallin eri objektit yhdistyvät toisiinsa. (Autodesk 3ds Max Help – Link Constraint.)

LookAt constraint kontrolloi objektin suuntaa siten, että se katsoo aina jotain toista objektia tai useampaa objektia kohti. Constraint lukitsee objektin rotaation siten, että objektin yksi akseli osoittaa suoraan kohti kohteena olevaa objektia. Viewportissa LookAt constraint näkyy sinisenä viivana, joka osoittaa katsottavan akselin suunnan. (Autodesk 3ds Max Help – LookAt Constraint.)

Orientation Constraint asettaa objektin orientaation seuraamaan kohteena olevan objektin tai objektien orientaatiota. (Autodesk 3ds Max Help – Orientation Constraint.)

Path Constraint rajoittaa objektin liikkeen seuraamaan polkua (spline) tai usean polun keskiarvoa. Path target voi olla minkä tahansa tyyppinen spline. Kohde polkua pystytään animoimaan samalla tavalla kuin muitakin objekteja. Jos polkua avainnetaan aliobjektitasolla, vaikuttavat sen muutokset myös siihen sidottuun objektiin. (Autodesk 3ds Max Help – Path Constraint.)

Position Constraintilla asetetaan objekti seuraamaan kohteena olevan objektin sijaintia (Autodesk 3ds Max Help – Position Constraint.) ja Surface Constraint rajoittaa objektin liikkeen toisen objektin pintaan. (Autodesk 3ds Max Help – Surface Constraint.)

### 2.5 Helpers

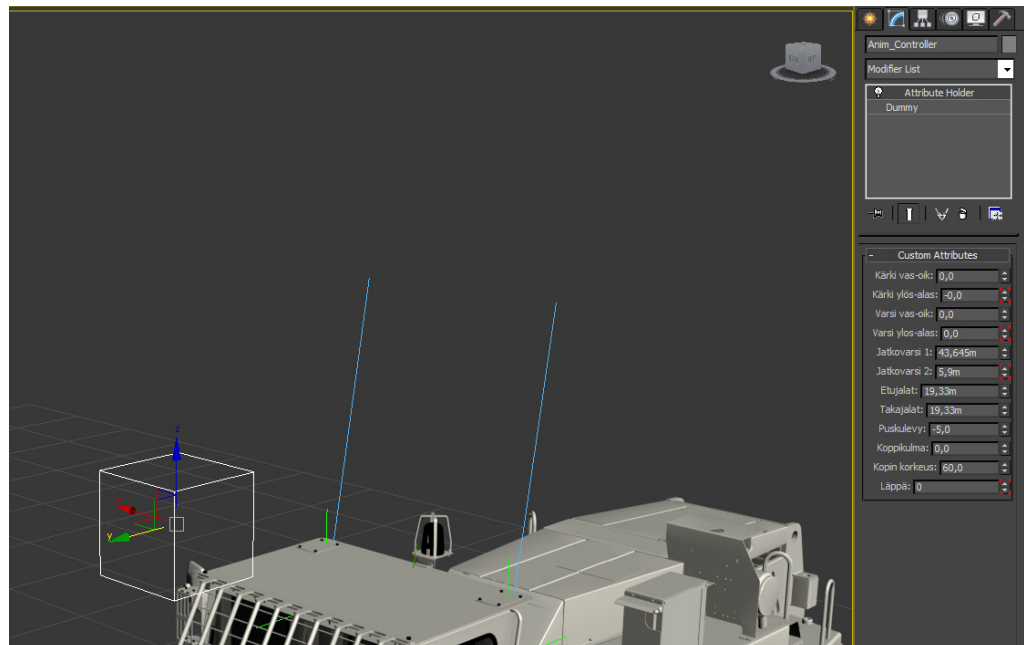
Helperssit ovat niin sanottuja dummy-objekteja, jotka eivät näy renderöitäessä ja niitä käytetään animaatioiden avustajina. Helpersien tehtävänä on avustaa monimutkaisten animaatioliikkeiden luomisessa sekä monimutkaisten hierarkioiden rakentamisessa. Koska dummyt ovat näkymättömiä renderöidyssä skenessä, ne toimivat hyvin vastapainonivelinä, objektien välisinä liittiminä sekä monimutkaisten hierarkioiden ohjaimina. (Autodesk 3ds Max Help – Dummy Objects.)

#### 2.5.1 Dummy helper

Dummy helper on wireframe-kuutio, jolla on pivot-piste sen keskipisteessä. Dummy helperillä on nimi, mutta sitä ei voi muokata, se ei näy renderöitäessä ja sillä ei ole omia parametreja.

Dummy helperiä käytetään avustamaan hierarkkista linkitystä ja dummyja käytetään usein animoimaan linkitettyä liikettä. Esimerkiksi jos dummy-objektia käytetään monen muun siihen linkitetyn objektin keskipisteenä, dummy objektia pyöritettäessä muut objektit kääntyvät sen mukana.

Dummy objektiin voidaan myös lisätä tiettyjä muuttujia, jotka auttavat animoitaessa. Esimerkiksi liittämällä dummyyn Attribute Holder-modifier, kuva 4, saadaan Wire Parameters- animaatiokontrollit asetettua omaan objektiin, jolloin animoijan ei tarvitse etsiä niitä muualta. (Autodesk 3ds Max Help – Dummy Helper.)

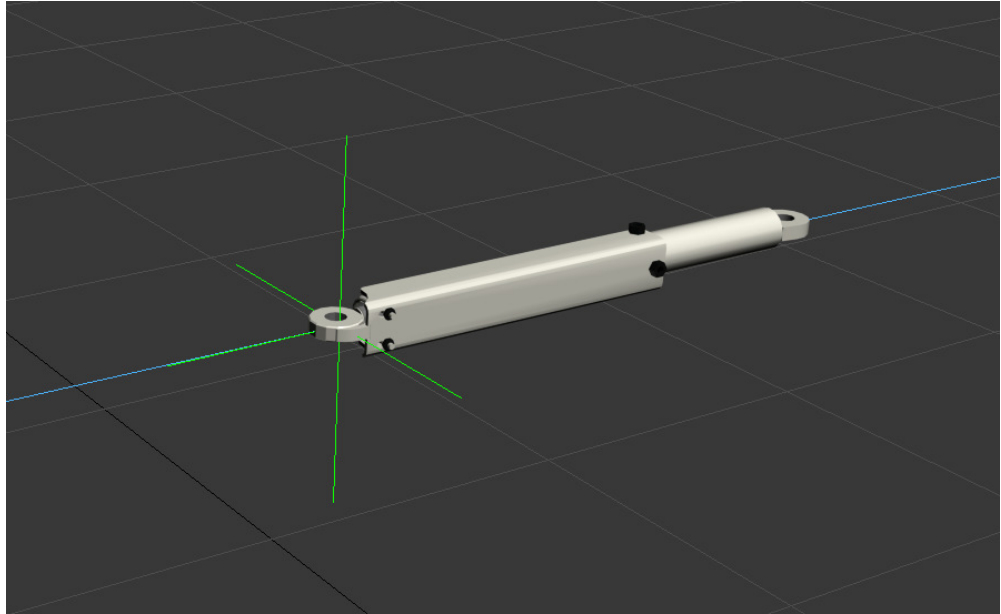


Kuva 4. Dummy-objekti, johon on liitetty Attribute Holder-muuttuja.

#### 2.5.2 Point Helper

Point antaa tietyn pisteen 3D-avaruudessa, jota voidaan käyttää referenssinä tai ohjelman muut funktiot voivat käyttää sitä hyödyksi

laskennassa. Rakennettaessa rigiä objektiin voidaan asettaa LookAt Constraint, joka katsoo tiettyyn pointtiin aina. Point on linkitetty johonkin toiseen objektiin, kuva 5. Kun toinen objekti liikkuu, point liikkuu sen mukana ja LookAt-objekti seuraa pointin liikettä. Tällöin voidaan varmistaa että rig ei hajoa animoitaessa, koska sen kaikki osat seuraavat toistensa liikettä sekä muuttavat omaa sijaintiaan ja asentoa sen mukaan. (Autodesk 3ds Max Help – Point Helper.)

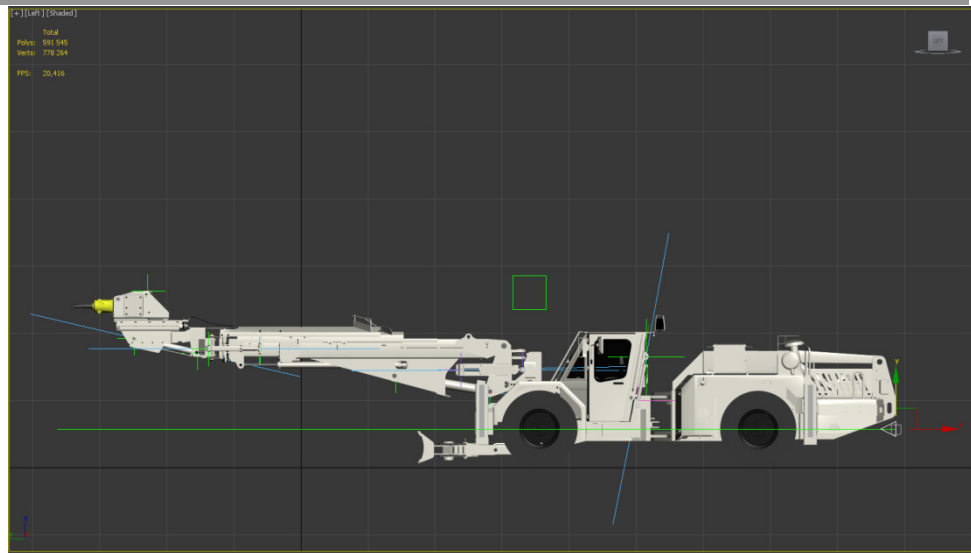


Kuva 5. Point helper.

### 2.5.3 Tape Helper

Tape on nimensä mukaisesti ruudulla näkyvä mittanauha, jota käytetään pituuksien mittaamiseen ja määrittämiseen. Tape rakentuu kahdesta objektista, itse tapesta ja sen target-objektista, kuva 6. Nämä kaksi objektia on yhdistetty viivalla, joka merkitsee niiden välistä pituutta.

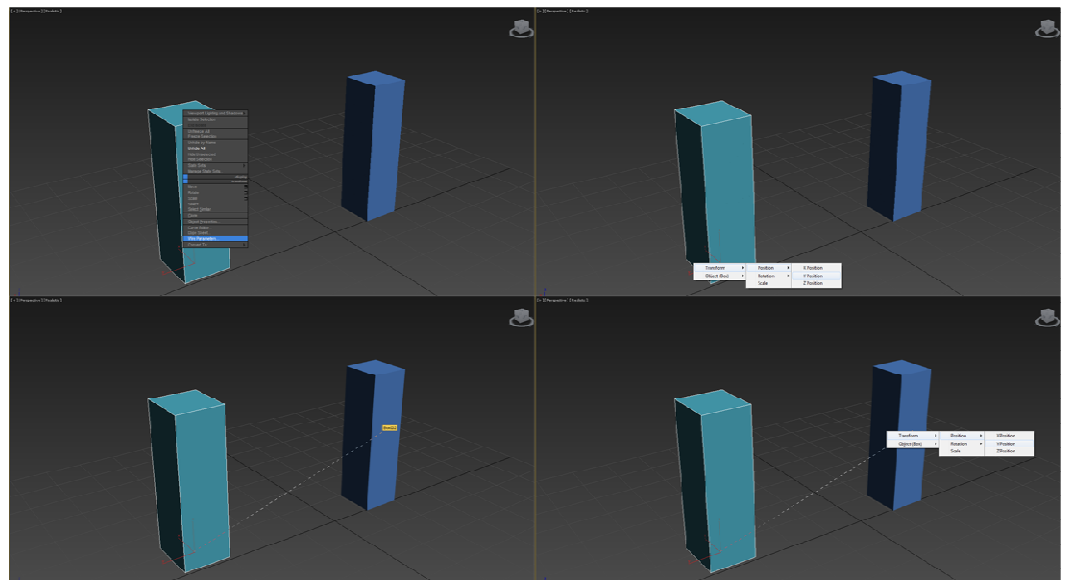
Rigiä rakennettaessa tape helpereitä käytetään varmistamaan että malli on oikean kokoinen ja että mallien osien liikkeiden pituudet ovat samanpituisia kuin reaali maailmassa. (Autodesk 3ds Max Help – Tape Helper.)



Kuva 6. Tape helper käytössä.

## 2.6 Wire Parameters

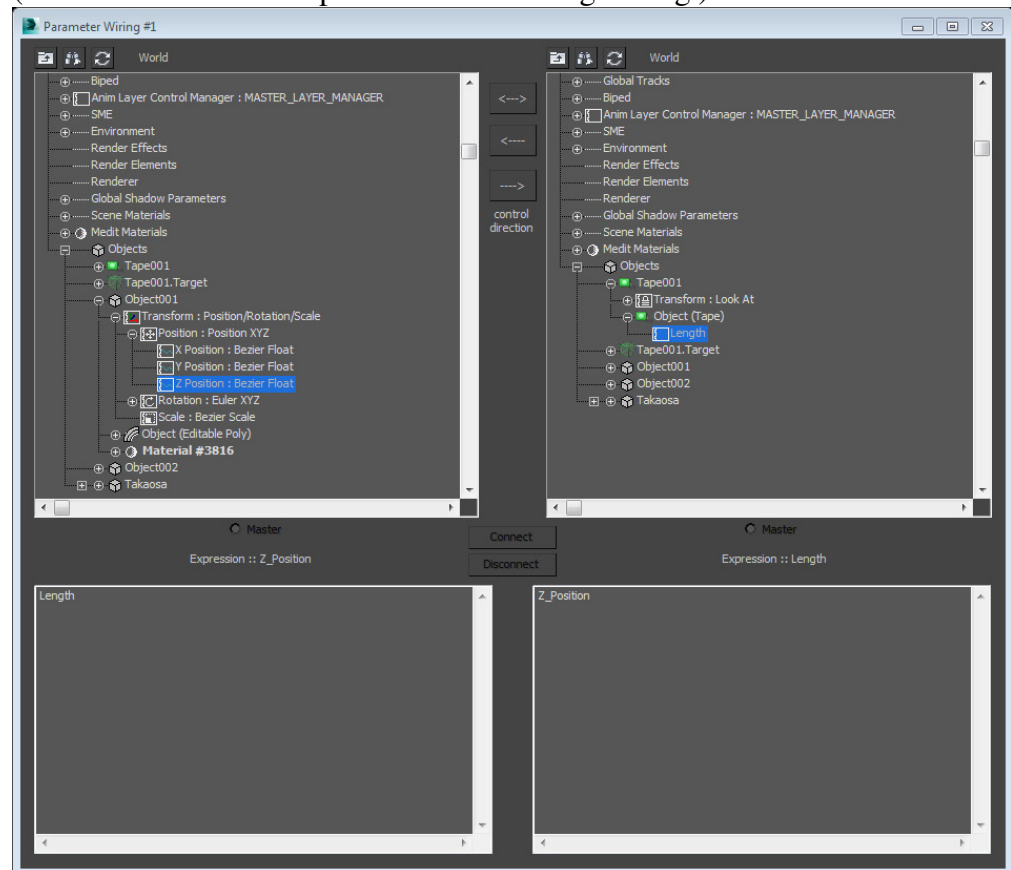
Wire Parameters mahdollistaa parametrien linkittämisen objektista toiseen. Tämä toimii suoraan viewportista, jolloin yhden parametrin muuttaminen muuttaa toista automaattisesti. Tällöin voidaan luoda yksi- tai kaksisuuntaisia yhteyksiä objektien tiettyjen parametrien välillä. Wire Parametersin avulla voidaan myös käyttää dummy-objekteja, jotka sisältävät halutut parametrit muiden objektien kontrolloimiseen, kuva 7. (Autodesk 3ds Max Help – Wire Parameters.)



Kuva 7. Wire Parameters linkitys viewportissa.

## 2.6.1 Parameter Wiring Dialog

Parameter Wiring ikkunassa voidaan muokata Wire Parametreiden suhteita. Ikkunassa voidaan luoda uusia yksi- tai kaksisuuntaisia kontrollisuhteita objektien parametrien (kuten sijainti, rotaatio, koko, materiaalit) välillä. Ikkunassa voidaan myös muokata jo olemassa olevia suhteita tai luoda ja muokata ilmaisuja, jotka määrittävät parametrien suhteita. Vain animoitavia parametrejä voidaan yhdistää toisiinsa ja sub-objektien pitää olla animoituja ennen kuin niitä voidaan yhdistää toisiin parametreihin. Parametriketjutukset voidaan myös yhdistää Custom Attributeista objekteihin, materiaaleihin ja modifiereihin, kuva 8. (Autodesk 3ds Max Help – Parameter Wiring Dialog.)



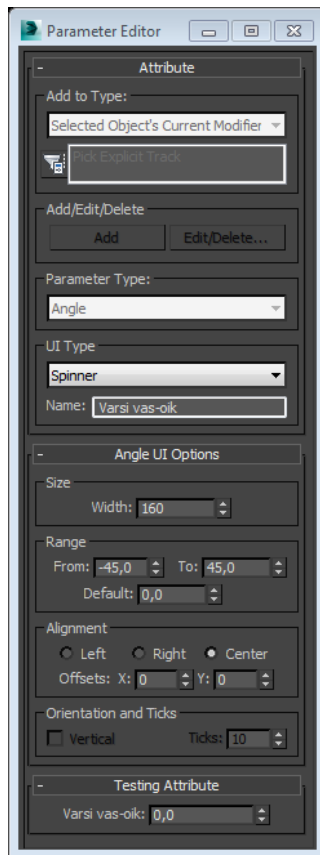
Kuva 8. Parameter Wiring-ikkuna.

## 2.7 Parameter Editor

Parameter Editoria käytetään liittämään ominaisuuksia objekteihin, modifiereihin, materiaaleihin ja animaatiopolkuihin. Ominaisuuksia ei ole niinkään määritelty, koska ne eivät suoraan vaikuta objektin toimintaan. Ominaisuudet vaikuttavat objekteihin vasta sen jälkeen, kun objektiin on asetettu wire parametrit tai kontrollit, jotka yhdistävät ominaisuuden toiseen parametriin skenessä. Ominaisuudet käyttäytyvät kuten muutkin objektiparametrit: niitä voidaan animoida ja ne tallennetaan ja ladataan objektin mukana työtiedostossa. (Autodesk 3ds Max Help – Custom Attributes.)



Jokainen ominaisuus voi olla jokin erityyppisistä datatyypeistä: kokonaisluku, arvo, totuusarvomuuttuja, taulukko, noodi, väri tai tekstuurikartta. Parametrit, jotka on lisätty objektiin tai modifieriin, näkyvät Custom Attributes-ikkunassa modify-paneelissa. Jokaiselle ominaisuudelle voidaan määrittää nimi, layout, arvoalue, oletusarvo ja käyttöliittymän tyyppi, kuva 9. Käyttöliittymän tyyppi voi olla spinner tai slider arvoille, valintaruutu totuusarvoille jne. (Autodesk 3ds Max Help – Custom Attributes.)



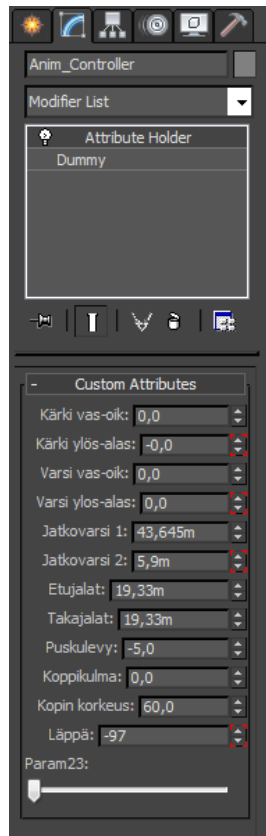
Kuva 9. Parameter Editor-dialogi.

## 2.8 Attribute Holder

Modifiereita käytetään muokkaamaan objekteja. Ne voivat muuttaa objektin geometriaa tai sen ominaisuuksia. (Autodesk 3ds Max Help – Modifiers.)

Attribute Holder-modifier on tyhjä muuttuja, joka luo Modify-paneelille käyttöliittymän, johon voi lisätä erilaisia määritteitä, jotka auttavat rigin liikkeiden hallinnassa, kuva 10. Attribute Holderilla ole omaa käyttöliittymää, vaan siihen itse lisätään määritteet, joita halutaan käyttää. Attribute Holder on riisutumpi versio Parameter Collectorista ja se näkyy vain modify-paneelissa eikä omana dialogi-ikkuna, kuten Parameter Collector. Attribute Holderiin luotuja parametrejä pystytään avaintamaan samalla tavalla kuin mitä tahansa muita parametrejä, joten animoija pystyy

tekemään monipuolisija liikkeitä animaatiokontrollien avulla.(Autodesk 3ds Max Help – Attribute Holder modifier.)



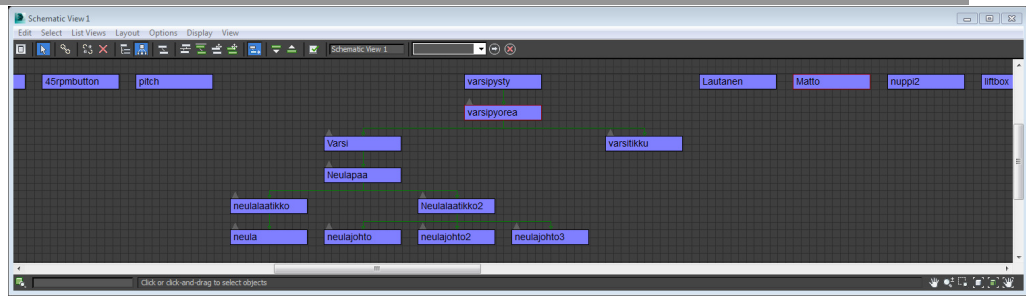
Kuva 10. Attribute Holder liitettyä dummy-objektiin.

### 2.9 Parameter Collector

Parameter Collector antaa työkalut järjestellä ja esittää animoitavat parametrit siten, että niitä pystytään käyttämään ja avaintamaan helposti. Parameter Collector on skaalautuva dialogi-ikkuna, joka päivittyy dynaamisesti parametrien muuttuessa. Dialogia voidaan järjestellä drag-and-drop-menetelmällä. Kerätyt parametrit tallentuvat skenejen mukana ja ne voidaan myös viedä toisiin skeneihin. Parameter Collectorin vahvuutena on mahdollisuus muuttaa kaikkien kerättyjen parametrien arvoja samanaikaisesti. (Autodesk 3ds Max Help – Parameter Collector.)

### 2.10 Schematic View

Rigattessa 3D-malleja saatetaan joutua käyttämään useita linkityksiä, constraintteja, parametreja sekä useita erilaisia suhteita objektien välillä. Tällöin on tärkeää saada selkeä kuva kokonaisuudesta, jotta tiedetään miten mikäkin osa on yhteydessä toisiinsa. Schematic View on näkymä, jossa näkyvät kaikki scenessä olevat osat kaaviossa. Schematic View-näkymästä nähdään objektien ominaisuudet, materiaalit, ohjaimet, määrittäet, hierarkia, ja muut skenen asetukset kuten linkitysasetykset, kuva 11. (Autodesk 3ds Max Help – Schematic View.)



Kuva 11. Hierarkia Schematic View-näkymässä.

Schematic View-näkymässä voidaan katsoa, luoda ja muokata objektien välisiä suhteita, sekä luoda hierarkioita, määrittää ohjaimia, materiaaleja, muuttujia sekä constrainteja. Näkymästä saadaan myös listanäkymä, jossa osat ja niiden ominaisuudet nähdään tekstimuodossa. Schematic Viewiin pystytään myös asettamaan taustakuva, jolloin noodien asettaminen on helpompaa, kun nähdään missä osat sijaitsevat fyysisesti ja voidaan sijoittaa noodit sen mukaisesti. (Autodesk 3ds Max Help – Schematic View.)

Jotta käyttäjän olisi helpompi nähdä miten Schematic Viewin objektit käyttäytyvät, ikkunassa on erilaisia käytänteitä objekteille, jotka ovat tietyissä tiloissa. (Autodesk 3ds Max Help – Schematic View.)

Kaikki objektit näytetään nimetyissä laatikoissa. Jos laatikon reuna on yhtenäinen, objekti on järjestetty. Jos reuna on avain, objekti on vapaa. Punaisella reunuksella osoitetaan, että objekti on animoitu. Jos laatikossa on nuoli lopussa, objekti jakaa suhteen jonkin toisen objektin kanssa. Jos laatikko on koko valkoinen, objekti valittuna Schematic Viewissä ja valkoinen reunus osoittaa objektin olevan valittuna viewportissa. Ylöspäin osoittava nuoli sulkee ja alaspäin osoittava nuoli avaa objektin child-objektit. (Autodesk 3ds Max Help – Schematic View.)

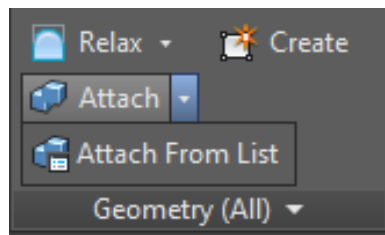
### 3 SCAMEC-MALLIN RIGGINGPROSESSIN TYÖNKULKU

Hämeen ammattikorkeakoulun mediaverstaan kesäprojektissa tuotettiin Normet Oy:lle visualisointi Scamec 2000 S laitteesta. Asiakas tarvitsi useita animaatioita laitteesta, jotka visualisoivat laitteen toimintaa. Animaatiota varten asiakas toimitti step-muotoisen CAD-mallin laitteesta sekä ohjekirjan laitteen toiminnasta.

#### 3.1 CAD-mallin valmistelu

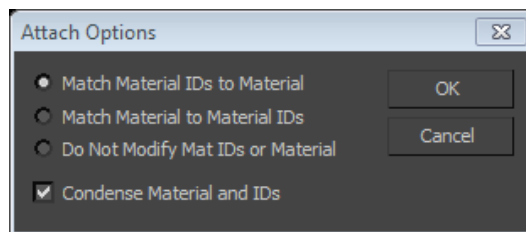
Ensin asiakkaalta saatu step-malli käännetään Autodeskin Inventor ohjelmalla polygonimuotoon, ja malli tuodaan Autodeskin 3ds Max-ohjelmaan. Tuodussa mallissa on satoja yksittäisiä osia ja yli miljoona polygonia, joten sen käsittely ohjelmassa on raskasta. Ennen riggaamisen aloittamista, laite ensin yhdistetään yhdeksi objektiksi ja sitten irrotetaan tarvittavat osat objektista omiksi objekteiksi. Tällöin riggaamisen toteuttaminen on huomattavasti helpompaa.

Laitteen eri osat saadaan yhdistettyä valitsemalla, jokin objekti ja sen jälkeen valitaan Attach From List- työkalurivin Geometry kohdasta, kuva 12.



Kuva 12. Attach From List-toiminto.

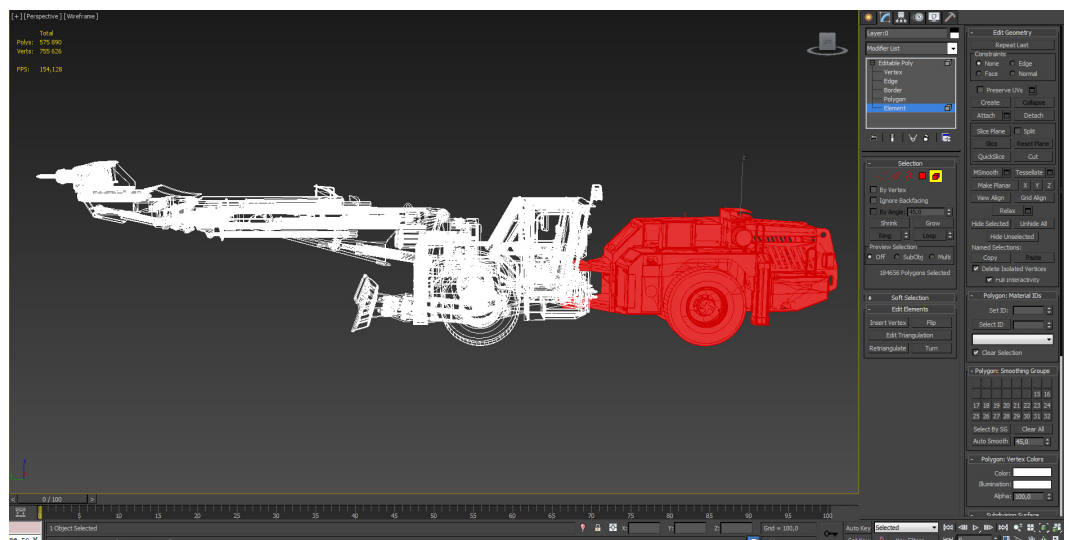
Avaustuvasta dialogista valitaan kaikki objektit ja hyväksytään valinta. Tämän jälkeen ohjelma kysyy halutaanko materiaalien ID:itä yhdistää. Tämä riippuu siitä, millaiset materiaalit saadussa mallista on: jos objekteissa on jo asetettu materiaalit, ei materiaaleja kannata yhdistää, kuva 13.



Kuva 13. Objektien yhdistämisen valinnat.

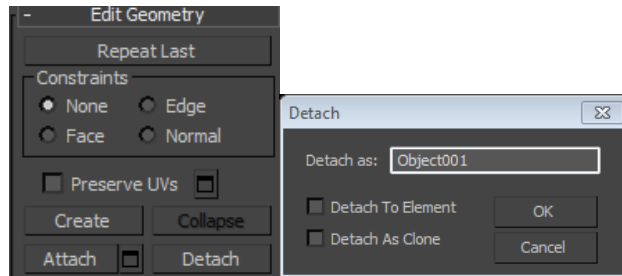
Kun valinnat on hyväksytty, ohjelma yhdistää objektit yhdeksi objektiksi. Prosessin kesto riippuu hyvin paljon objektien määrästä: mitä enemmän objekteja, sitä enemmän aikaa niiden yhdistämiseen kuluu.

Kun laite on yhdistetty yhdeksi objektiksi, voidaan aloittaa laitteen jakaminen pienemmiksi, helpommin käsiteltäviksi osiksi. Tämä toteutetaan siten, että sub-objektitasolla valitaan elementit, jotka kuuluvat yhteen, kuva 14.



Kuva 14. Elementtitason valita

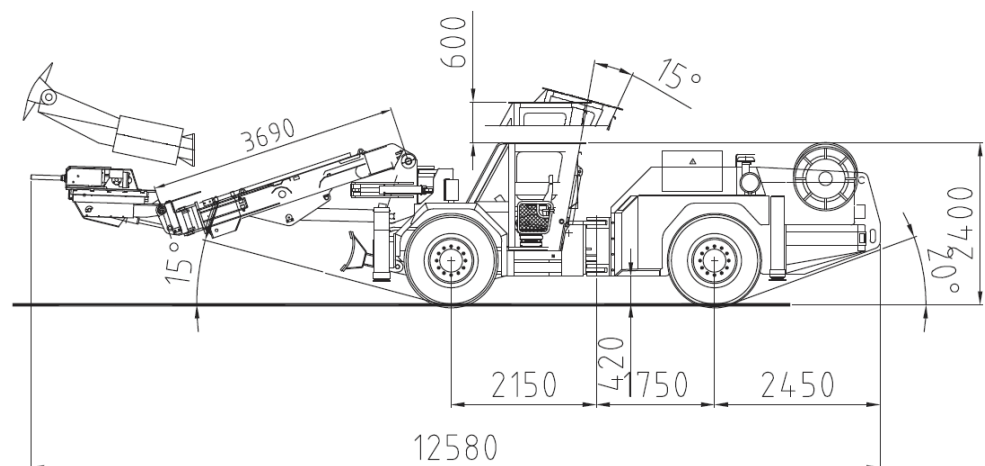
Kun valinta on tehty, käytetään modify-paneelista löytyvää detach-toimintoa, joka irrottaa valitut elementit omaksi objektikseen. Detach-toimintoon kuluva aika riippuu siitä, kuinka monta objektia on valittuna. Detach dialogi kysyy objektin nimeä, irrotetaanko osa elementiksi sekä kloonataanko irrotettu osa, kuva 15.



Kuva 15. Detach-näppäin ja valinnat

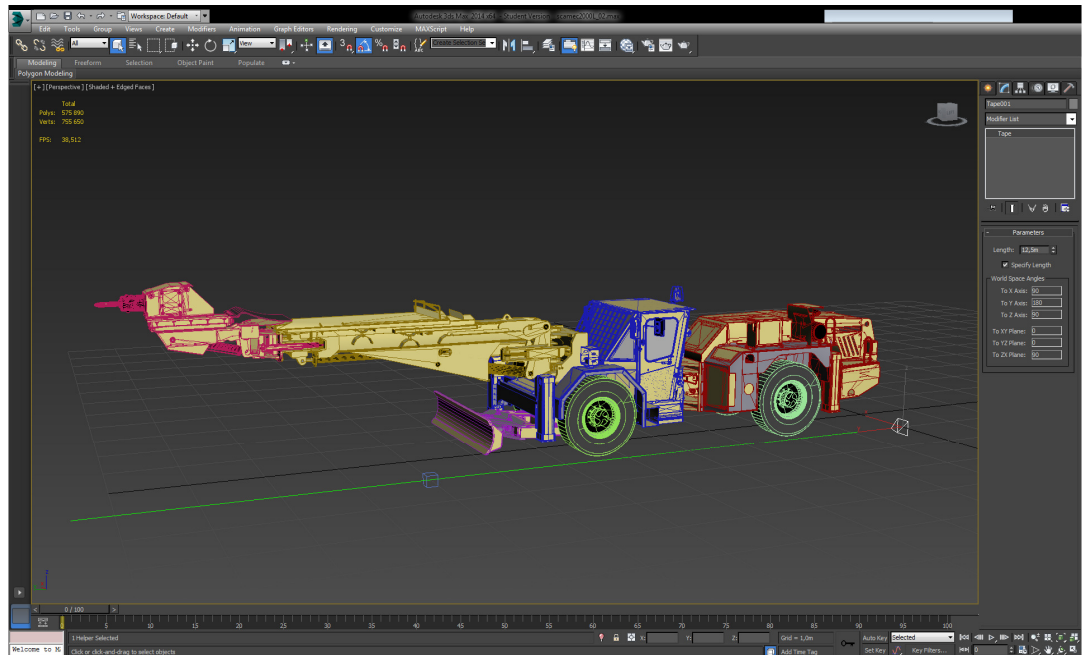
Valinnat tehdään siten, että ne osat jotka liikkuvat muista osista riippumattomina, irrotetaan omiksi objekteiksi, jotta animaatiota varten tehtävä rigging olisi mahdollista toteuttaa.

Scamec laitteen paloittelu aloitetaan jakamalla objekti loogisiin elementteihin. Valintojen tekemisessä käytetään avuksi asiakkaan antamaa laitteen spesifikaatio-ohjetta, jossa näkyy mitkä osat laitteesta liikkuvat ja millä tavalla. (Normet, 1-7, kuva 16)



Kuva 16. Ote Scamec-spesifikaatiosta. (Normet, 5)

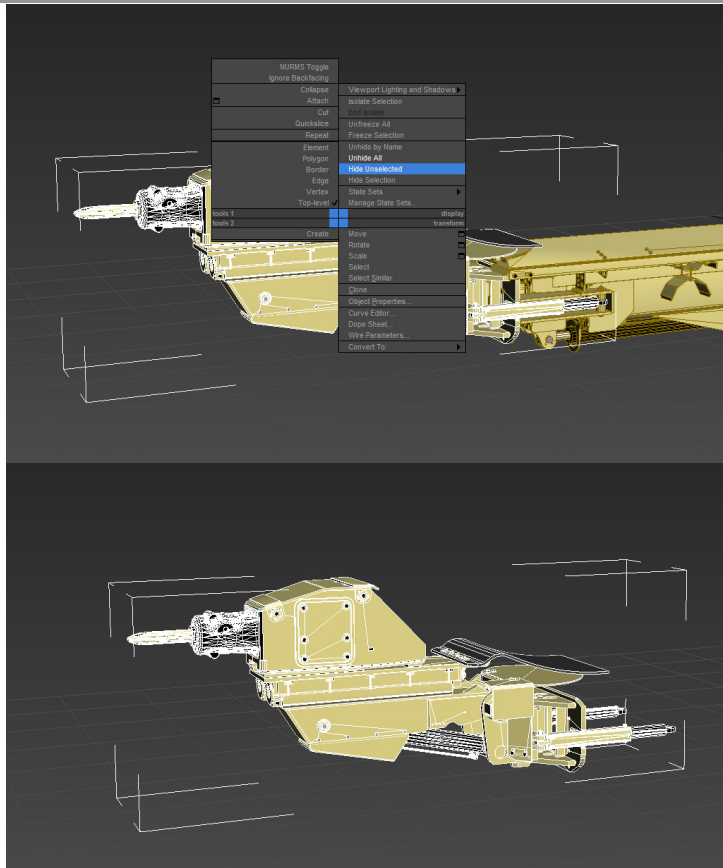
Laite on toiminnaltaan symmetrinen, joten samaa valintaa voidaan käyttää molemmilla puolilla. Scamec laite jaetaan ensin kahdeksaan eri osaan: takaosa, keskiosa, puskulevy, kärkiosa, varsi, eturenkaat ja takarenkaat, kuva 17.



Kuva 17. Edge-näkymässä nähdään eri osiin jaettu laite.

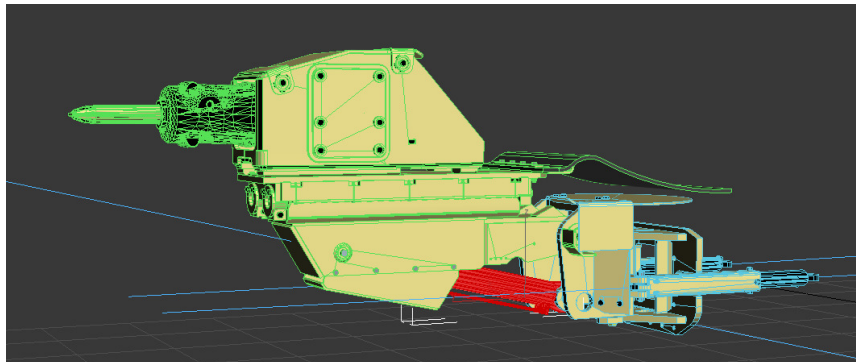
Tämän jälkeen voidaan aloittaa yksittäisten objektien jakaminen vielä pienempiin osiin. Esimerkki tapauksena käytetään kärkeosan paloittelua, mutta samaa tekniikkaa hyödynnetään myös koneen muihin osiin.

Ensin valitaan käytettävä objekti ja piilotetaan muut objektit käyttämällä Hide unselect-toimintoa, jotta vain valittu objekti jää näkyville. Tämä helpottaa objektien käsittelyä, kuva 18.



Kuva 18. Hide unselected-toiminto.

Valitun objektin modifier-paneelissa siirrytään elementti sub-objektitasolle ja valitaan osat jotka halutaan irrottaa. Irrottamiseen käytetään detach-toimintoa samalla tavalla kuin aiemmassa vaiheessa, kuva 19.



Kuva 19. Objektin valinta elementti sub-objektitasolla.

Kun objektista on irrotettu kaikki liikkuvat osat omiksi objekteiksi, voidaan siirtyä objektien hierarkkiseen linkittämiseen ja riggaamiseen. Tämä vaihe tehdään laitteen jokaiselle objektille, jotta voidaan siirtyä rigging-vaiheeseen ja sen jälkeen animointikontrollien luomiseen.

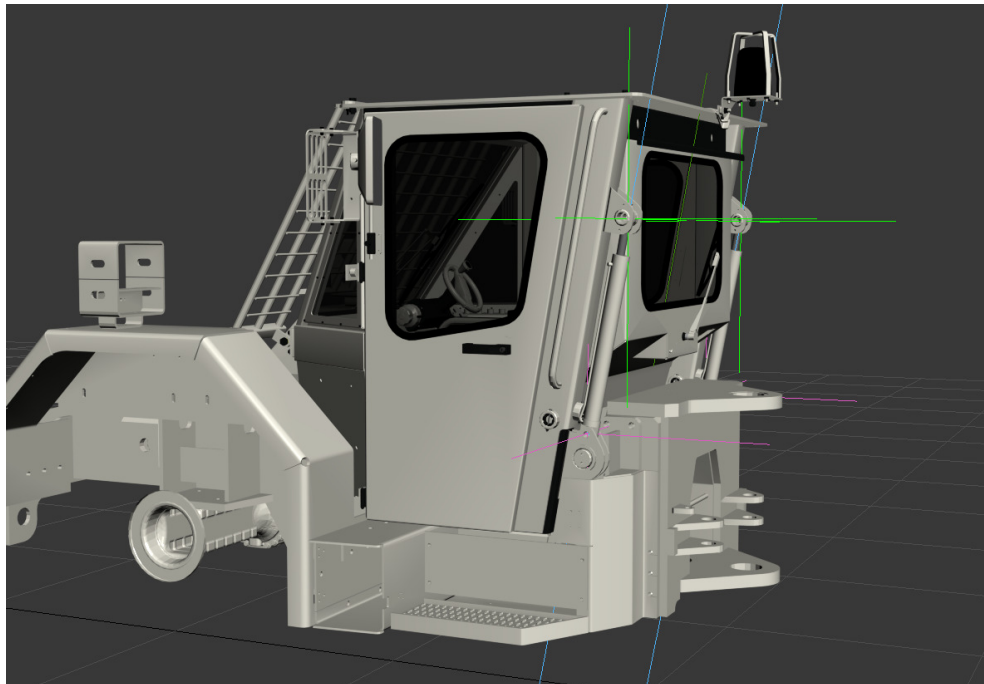
### 3.2 Riggaaminen

Kun laitteen osat on jaettu helpommin käsiteltäviin osiin, voidaan siirtyä rigging-vaiheeseen. Kappaleessa käsitellään laitteen riggingissä käytettäviä tekniikoita, joita sovelletaan laitteen eri osissa, muttei käydä laitteen jokaisen osan riggingprosessia läpi erikseen.

#### 3.2.1 Linkittäminen

Laitteen rigginssä käytetään hierarkkista rakennetta, jossa osat linkittyvät toisiinsa parent/child-suhteessa. Laitteen osat voidaan linkittää ensin omiksi hierarkkisiksi puurakenteiksi ja sen jälkeen yhdistää hierarkiat yhdeksi isoksi hierarkiaksi. Linkitys voidaan toteuttaa joko suoraan viewportissa käyttäen Select And Link-työkalua tai Schematic View-ikkunassa.

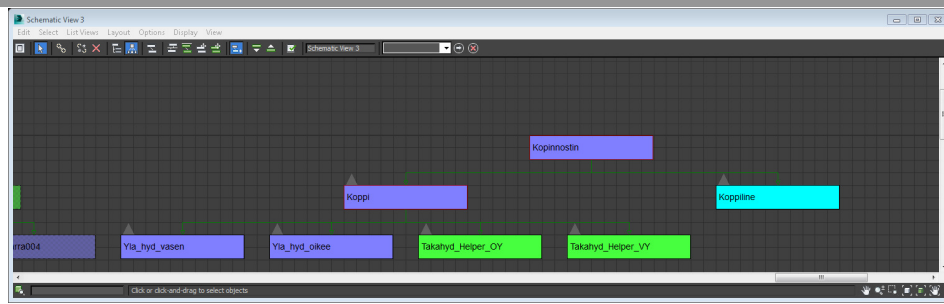
Esimerkkitapauksena kopin linkittäminen. Koppi rakentuu hydraulikkaosista, itse kopista ja kopinnostimesta. Koppiin on lisätty myös helper-objekteja sekä spline-polku, jonka mukaan koppi liikkuu, kuva 20.



Kuva 20. Scamec-laitteen koppi.

Aloitetaan objektien linkittäminen child-objekteista parent-objekteihin. Koska hydraulikan yläosat sijaisevat fyysisesti kiinni itse kopista, linkitetään ne yksi kerrallaan koppiin. Myös hydraulikan helper-objektit linkitetään koppiin. Koppi on kiinni kopinnostimessa, joten koppi linkitetään kopinnostimeen. Myös koppia ohjaava spline linkitetään kopinnostimeen, koska se ohjaa kopin liikkeitä, jolloin sen pitää myös seurata muun laitteen sijaintia.





Kuva 21. Kopinnostimen hierarkkinen rakenne.

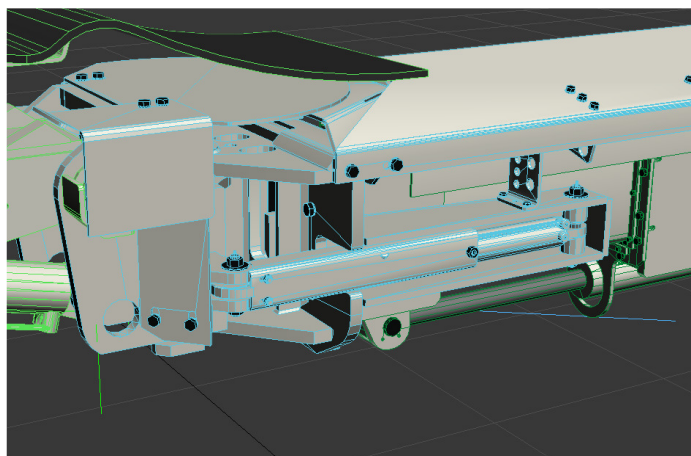
Koska kopinnostin ja hydraulikan alaosat sijaisevat laitteen keskiosassa, linkitetään nämä objektit keskiosaan. Tällöin kopin hierarkia yhdistyy koko laitteen kattavaan hierarkiaan, kuva 21.

Tätä tekniikkaa sovelletaan myös muun laitteen linkittämisessä. Tärkeintä on seurata hierarkkisen linkityksen periaatteita. Hyvin suunniteltu linkitys säästää niin riggaajan kuin animoijankin aikaa. Kun linkitykset on toteutettu voidaan siirtyä objektien kontrollien ja constraintien asettamiseen.

### 3.2.2 Ohjaimet ja constraintit

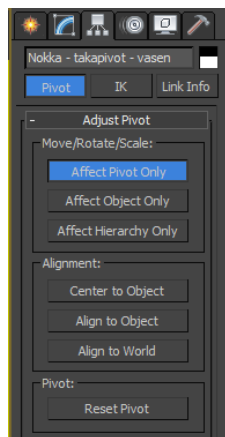
Jotta animaatiokontrollit voidaan luoda, joudutaan linkittyihin objekteihin usein lisäämään ohjaimia ja constrainteja, jotta animaatiot toimisivat halutulla tavalla. Koska hierarkkisessa IK-rakenteessa child-objektien muutokset vaikuttavat parent-objekteihin, tulee niiden liikettä rajoittaa constraintien avulla siten, että objektit liikkuvat oikealla tavalla.

Esimerkki tapauksena Scamec-laitteen kärkisosan hydraulikan asettaminen käyttämällä helper-objekteja, kontrolleja sekä constrainteja. Kärkisosan hydraulikkaosat koostuvat kahdesta objektista, jotka on linkitetty niihin kuuluviin rungon osiin, kuva 22. Koska animoitaessa osat liikkuvat x,y ja z-suunnassa ja rotaatiossa, joudutaan osiin asentaa kontrolleja, jotta laitteen osat liikkuvat yhdenmukaisesti animaatiossa.



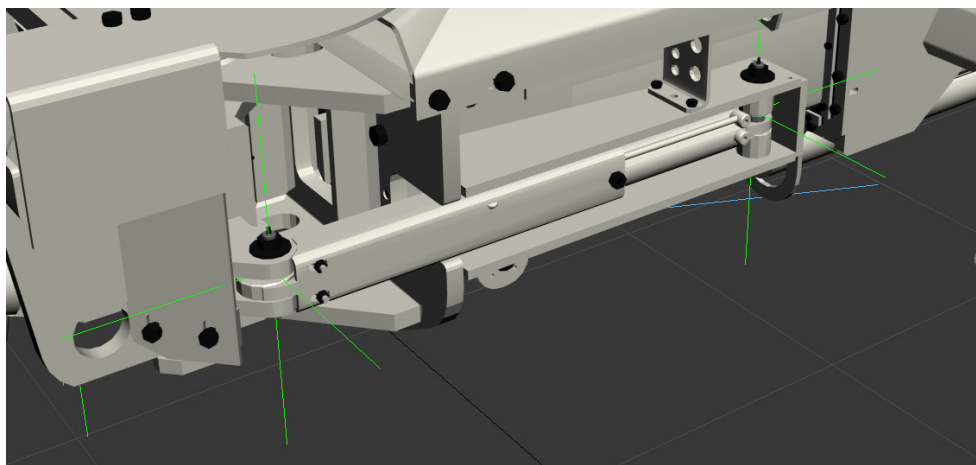
Kuva 22. Scamec-laitteen hydraulikka.

Koska objektien rotaatio tapahtuu niiden juuresta, aloitetaan siirtämällä objektien pivot-pisteet paikoilleen. Valitsemalla valitun objektin motion-paneelistä Adjust Pivot-kohdasta Affect Pivot Only, voidaan objektin pivot-pistettä siirtää ilman, että se vaikuttaa muuhun objektiin, kuva 23.



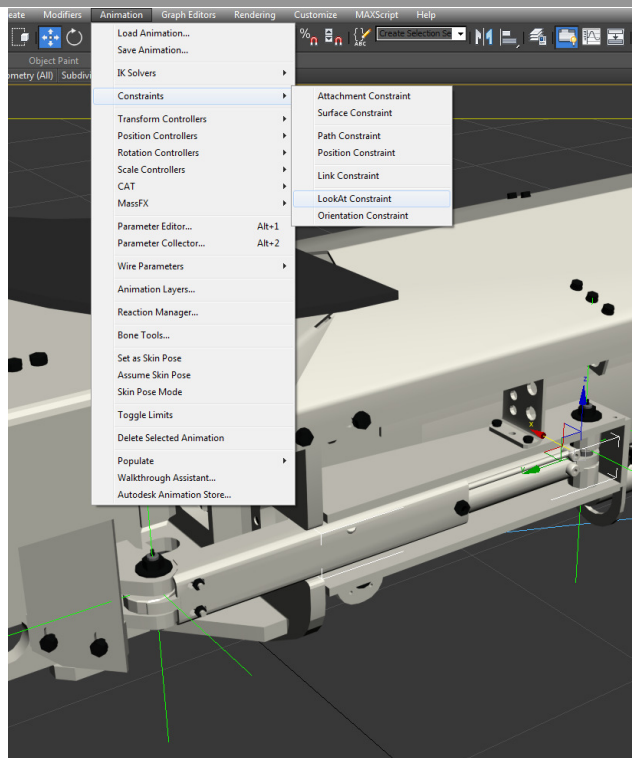
Kuva 23. Pivot-pisteen asettaminen.

Kun pivot-pisteet on asetettu paikoilleen, lisätään animoitavan hydrauliiikan molempien pivot-pisteiden kohdalle Point helper-objektit. Point helpereiden tehtävänä on ohjata animaatioita asettamalla objekteihin LookAt constraintit, jotka seuraavat Point helpereiden liikettä. Point helperit linkitetään samaan runkoon kuin hydrauliiikan osat, kuva 24.



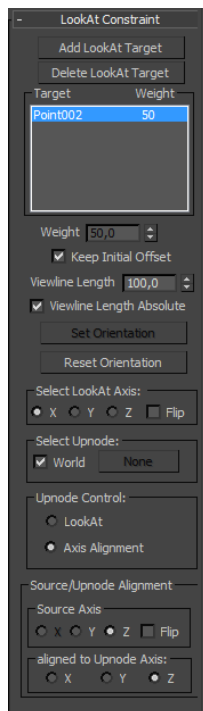
Kuva 24. Pivot-pisteiden kohdalle asetetut Point-helperit.

LookAt constraintit asetetaan valitsemalla objekti, valitsemalla Animation-> Constrains -> LookAt Constraint ja valitsemalla objekti johon halutaan katsoa, tässä tapauksessa vastakkainen Point helper. LookAt constraint voidaan myös asettaa objektin motion-paneelin Assign Controller kohdasta. LookAt Constraintin asettaminen tehdään myös toisen puolen objektille, jolloin molemmat osat seuraavat toistensa liikettä, kuva 25.

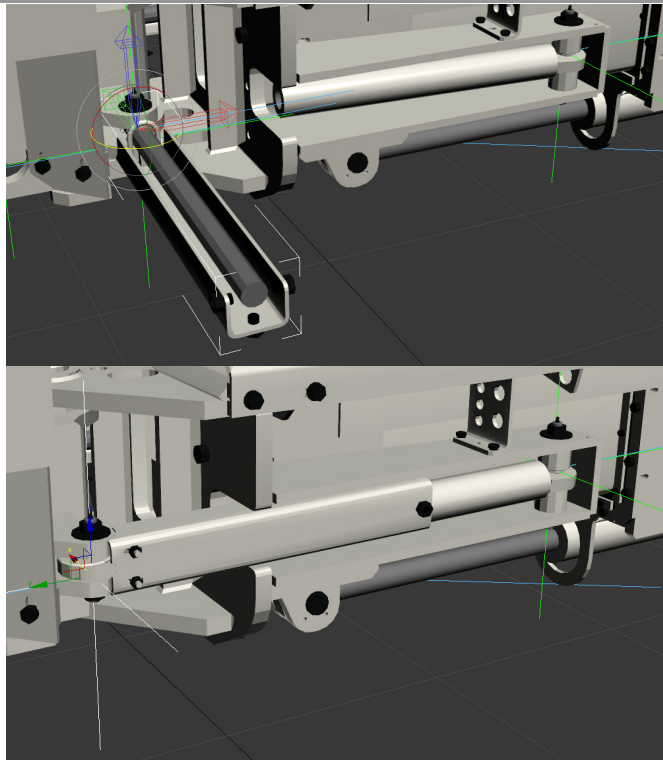


Kuva 25. LookAt Constraintin asettaminen valikosta.

Kun LookAt Constraint asetetaan saattaa objekti kääntyä väärään asentoon. Tämä korjataan muokkaamalla objektin pivot-pistettä. Valitaan Motion-paneelin PRS Parameters (eli sijainnin, rotaation, skaalauksen parametrit) kohdasta Rotation valinnat, joka avaa LookAt Constraintin ominaisuudet. Valitaan Select LookAt Axis ja Source Axis siten, että pivot-pisteiden asento vastaavat toisiaan, kuvat 26-27.

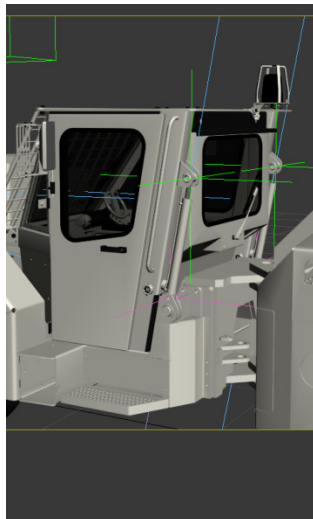


Kuva 26. LookAt Constraintin ominaisuudet Motion-paneelissa.



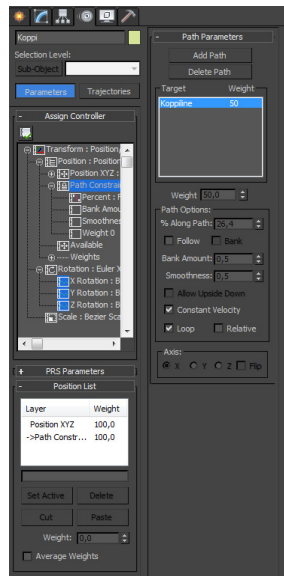
Kuva 27. Objektin asennon korjaaminen.

Toinen esimerkki on kopin liikkeen rajoittaminen seuraamaan polkua. Kopin hydraulikat on linkittetty ja kontrolloitu samalla tavalla kuin ensimmäisessä tapauksessa. Koska koppi liikkuu myös z-akselilla rotaation lisäksi, lisätään koppiin Path constraint-kontrolleri kopin liikkeen rajaamiseksi, kuva 28.



Kuva 28. LookAt Constrainteilla kontrolloitu laitteen koppi.

Luodaan shapes-paneelistä viewporttiin yksi kopinnostimen suuntainen spline-polku ja asetetaan se kopin keskelle. Tehty spline linkitetään kopinnostimeen. Seuraavaksi valitaan koppi-objekti ja lisätään siihen motion-paneelistä path constraint-kontrolleri position-listaan. Asetetaan seurattavaksi poluksi tehty spline. Nyt kopin sijanti ja kulma seuraa splineä, joka on linkitetty kopinnostin-objektiin, kuva 29.



Kuva 29. Path Constraintin asettaminen objektiin.

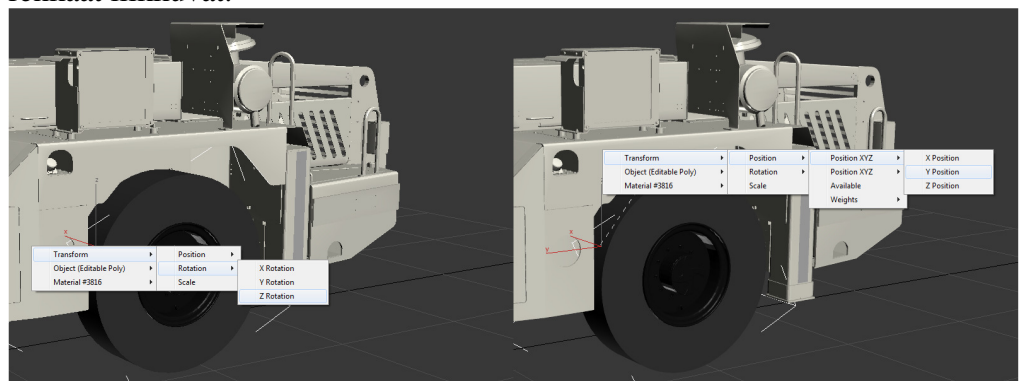
Nyt objektit seuraavat toisiaan, jolloin on mahdollista toteuttaa animaatiokontrollit. Kun objektit on linkitetty ja kontrollit asennettu, IK-ketjun animointi vaikuttaa koko laitteeseen. Näitä tekniikoita sovelletaan myös muun laitteen valmisteluun animaatiota varten.

### 3.2.3 Renkaiden pyöriminen

Riggattaessa laitetta on usein myös sellaisia osia, joita ei animoida käsin, vaan niiden pitää toimia suoraan muun animaation mukana. Esimerkki tällaisista osista on renkaat.

Koska renkaiden pitää liikkua muun laitteen mukana, linkitetään ne siihen osaan laitetta, jossa ne sijaitsevat fyysisesti. Scamec-laitteessa eturenkaat linkitetään keskiosaan ja takarenkaat takaosaan. Linkitys tehdään käyttäen Select and Link-toimintoa viewportissa.

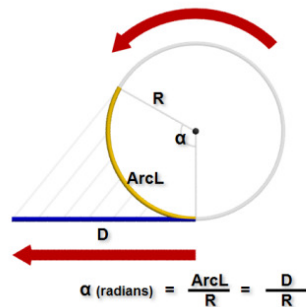
Kun linkitys on tehty, parameter wiring dialogissa asetetaan renkaiden Z-akselin rotaatio seuraamaan siihen linkitetyn osan Y-akselin suuntaista liikettä, kuva 30. Tällöin laitteen liikuessa eteenpäin myös laitteen renkaat liikkuvat.



Kuva 30. Renkaan z-rotatation linkittäminen takaosan y-positioon.

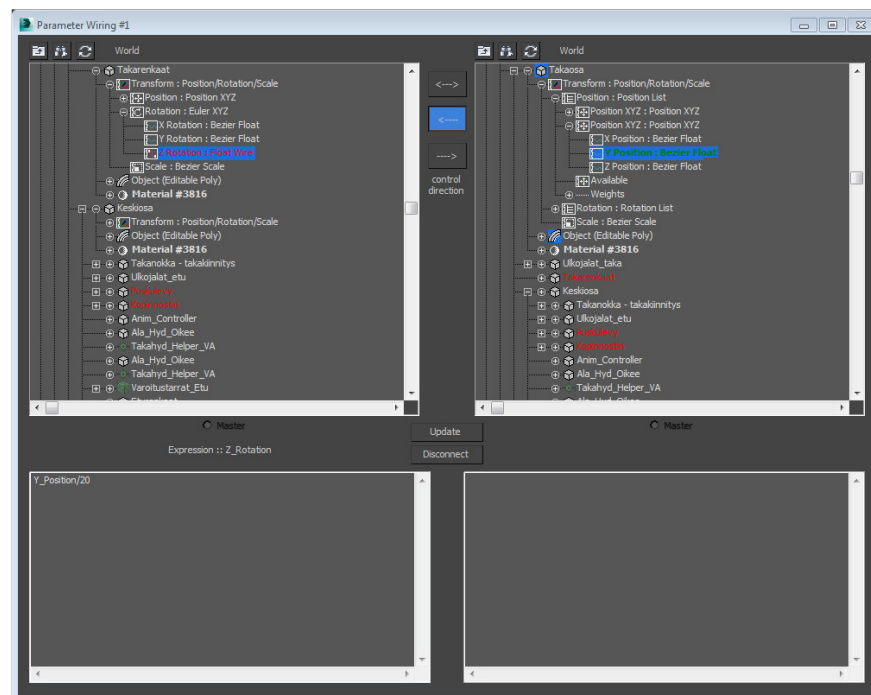
Koska renkaiden on pyörittävä oikeassa suhteessa laitteen liikkeeseen nähden, on linkitykseen asetettava ilmaisut, jotka määrittävät, kuinka paljon rengas kääntyy laitteen liikkeeseen verrattuna.

Pyöreän objektin rotaatio  $\alpha$  määrittyy ympyrän säteen ja kehän pituuden määrittämän kulman perusteella. Tämä rotaatio radiaaneina on yhtä kuin säteen pituus jaettuna säteellä. Tällöin säteen pituus on vakio ja kehän pituus suoristettuna on sama kuin auton renkaiden kulkema matka. Tällöin renkaiden rotaation kaava, kehän pituus jaettuna säteellä on yhtä kuin kuljettu matka jaettuna säteellä, kuva 31. (Autodesk 3ds Max Tutorials – Rotating The Wheels.)



Kuva 31. Ympyrän kehän kulkema matka. (Autodesk 3ds Max Tutorials – Rotating The Wheels.)

Scamec laitteessa renkaan säde on 20 yksikköä, lisätään renkaiden Z-rotatation ilmaisuun  $Y\_Position/20$ , jolloin laitteen liikkuessa x matkan, renkaat kääntyvät z-akselilla  $x/20$  verran, kuva 32.

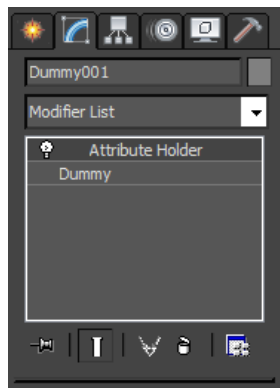


Kuva 32. Z-rotatation ilmaisu.

### 3.2.4 Animaatiokontrollien asettaminen

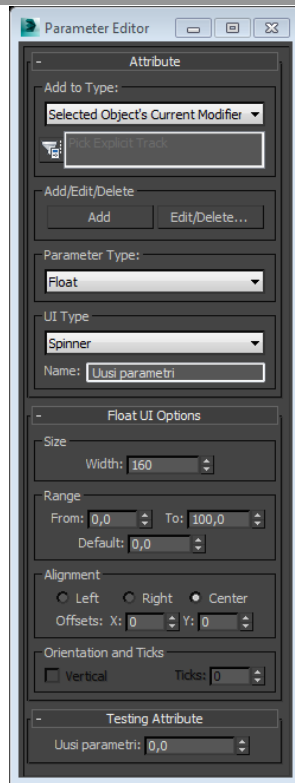
Kun objekteille on asetettu linkitykset ja niiden kontrollit, voidaan aloittaa animaatiokontrollien luominen. Animaatiokontrollien tarkoituksena helpottaa animoijan työprosessia, luomalla Custom Attributes-käyttöliittymä objektien sijainnin, rotaation ja koon hallinnalle. Custom Attributes voidaan toteuttaa joko lisäämällä Attribute Holder-modifier objektiin tai käyttämällä Parameter Collectoria.

Luodaan uusi dummy-objekti ja lisätään siihen Attribute Holder-modifier modify-paneelissa. Attribute Holder voidaan myös lisätä mihin tahansa muuhun objektiin skenessä, mutta dummy-objektiin asetettu modifier helpottaa animoijan työprosessia, koska se voidaan pitää erillään muista objekteista. Attribute Holder ei sisällä suoraan käyttöliittymäominaisuuksia, vaan riggaajan on luotava ne itse käyttäen Parameter Editoria, kuva 33.



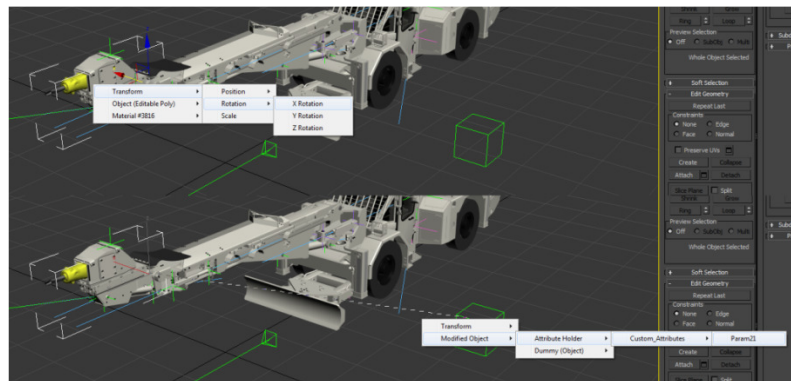
Kuva 33. Tyhjä Attribute Holder-modifier asetettuna dummy-objektiin,

Avataan Parameter Editor Animation-valikosta (Alt+1) ja valitaan luotu dummy-objekti. Parametrit lisätään objektin sillä hetkellä valittuna olevaan modifieriin, eli Attribute Holderiin. Seuraavaksi asetetaan parametrin tyyppi, käyttöliittymän tyyppi sekä parametrin nimi. Parametrien tyypit (kokonaisluku, arvo, totuusarvomuuttuja, taulukko, noodi, väri tai tekstuurikartta) vaikuttaa käytössä oleviin käyttöliittymäominaisuuksiin. Luotua attribuuttia voi kokeilla Testing Attribute- kohdasta, jolloin nähdään toimiiko ominaisuus halutulla tavalla. Parametrit lisätään Add- painikkeella ja luotuja parametreja voidaan muokata Edit/Delete- painikkeesta avautuvasta dialogi-ikkunasta, kuva 34.



Kuva 34. Parameter Editorin ominaisuudet.

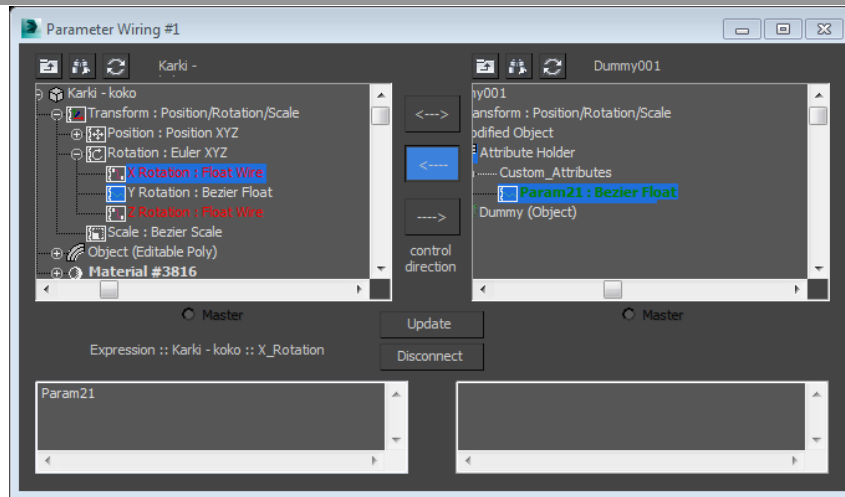
Kun uusi parametri on lisätty Attribute Holderiin, linkitetään jokin objekti kyseiseen parametriin. Valitaan objektin wire parameters valikosta haluttu parametri. Parametri linkitetään dummy-objektin Attribute Holderissa olevaan parametriin, joka luotiin aiemmin, kuva 35.



Kuva 35. Objektin parametrin linkittäminen luotuun parametriin.

Kun linkitys on asetettu, avautuu Parameter Wiring dialogi, kuva 36. Dialogissa valitaan parametrin kontrollointisuunta parametrin objektiin ja yhdistetään linkitys connect-painikkeella. Tämän jälkeen Attribute Holderissa sijaitsevalla käyttöliittymällä voidaan kontrolloida linkitetyn objektin liikettä.





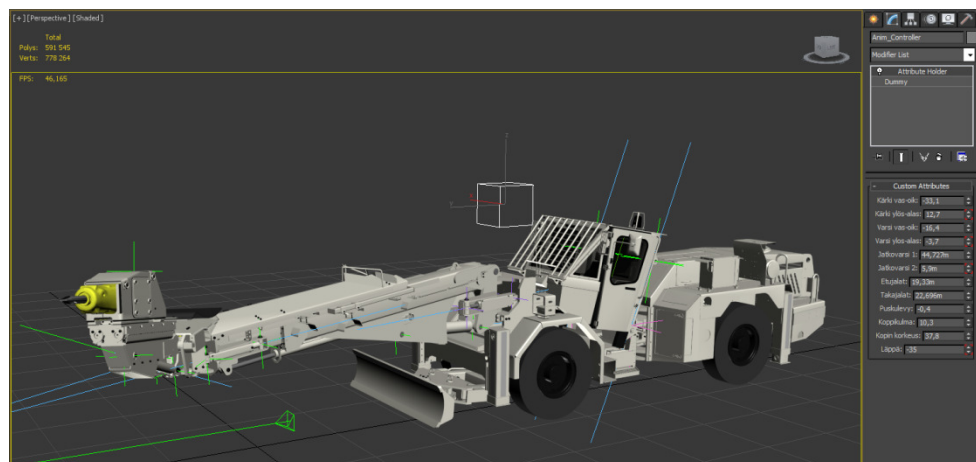
Kuva 36. Dialogissa asetettu parametrien kontrollisuunta ja ilmaisut.

Tämän jälkeen voidaan vielä muokata luodun parametrin ominaisuuksia Parameter Editorissa, jos alun perin luotu parametri ei toimi halutulla tavalla. Esimerkiksi rotaation raja-arvoja voidaan tarkentaa, jotta objekti kääntyy vain tiettyjen rajojen sisällä.

Tätä tekniikkaa sovelletaan laitteen kaikkien animoitavien osien animaatiokontrollien luomisessa. Kontrollit ovat kuitenkin tapauskohtaisia, joten yhtä oikeaa tapaa kontrollien luomiseen ei ole, vaan riggaajan täytyy itse tietää millaisia kontrolleja tarvitaan.

### 3.3 Valmis malli

Kun laiteelle on tehty linkitykset, luotu kontrollit ja constraintit, määritetty ei kontrolloitavat animaatio-osat sekä luotu käyttöliittymä animaatiokontrolleille, on laite valmis luovutettavaksi animoijalle, kuva 37. Animoija luo keyframe animaatiot käyttäen animaatiokontrolleja, minkä jälkeen luodaan valaistus ja kamerat, sekä renderöidään lopullinen video.



Kuva 37. Rigattu laite, jonka asentoa on muutettu käyttämällä animaatiokontrolleja.

#### 4 YHTEENVETO

Koska jokainen 3D-malli on erilainen, voi rigging-prosessi olla todella yksinkertainen tai hyvin kompleksi. Riggaajalla tarvitsee olla syvä tietämys käytettävistä sovelluksista ja niiden ominaisuuksista sekä tietoa skriptauksesta ja ilmaisujen luomisesta, koska työ on hyvin teknistä. (Beane 6-1.) Hyvin suunnitellut kontrollit säästävät animoijan aikaa, joten riggingin onnistuminen on 3D-visualisointiprojektin toteutumisen kannalta erittäin tärkeää.

Riggaajan tulee pystyä soveltamaan eri tekniikoita erilaisiin malleihin ja luotujen animaatiokontrollien tulee olla tarpeeksi kattavia, jotta animaatiot ovat mahdollisia toteuttaa, mutta tarpeeksi yksinkertaisia, että animoija pystyy suorittamaan tehtävänsä mahdollisimman helposti.

Scamec-projektissa onnistunut rigging ja hyvin toteutetut animaatiokontrollit säästivät huomattavasti aikaa laitteen animoinnissa ja renderöintivaiheeseen siirtyminen tapahtui hyvissä ajoin. Koska asiakas tarvitsi korjauksia valmisiin renderöinteihin, oli tärkeää pystyä tekemään muutoksia animaatioihin helposti, jotta uudet renderöinnit saatiin tehtyä aikataulun sisällä.

Projektissa kehitettiin jo opittuja tekniikoita ja hankittiin runsaasti uutta tietotaitoa, jonka avulla tulevien 3D-visualisointien toteuttaminen Hämeen ammattikorkeakoulun mediaverstaalla kehittyy. Jatkossa pystytään vielä kehittämään uusia, paranneltuja tekniikoita rigging-prosessiin, koska 3D-ohjelmat kehittyvät jatkuvasti.

## LÄHTEET

Autodesk 3ds Max Help 2014

Autodesk 3ds Max Tutorials 2014

Beane, A. 2012. 3D Animation Essentials. Indianapolis, Indiana: John Wiley & Sons.

Normet. 2013. Scamec 2000 S Technical data sheet. Viitattu 9.9.2013  
[http://www.normet.com/inet/normet/flow.nsf/%28AttachmentsByFileName%29/Scamec%202000%20S%20100043782.pdf/\\$File/Scamec%202000%20S%20100043782.pdf](http://www.normet.com/inet/normet/flow.nsf/%28AttachmentsByFileName%29/Scamec%202000%20S%20100043782.pdf/$File/Scamec%202000%20S%20100043782.pdf)

Vasconcelos, V. 2011. Blender 2.5 Character Animation Cookbook. Iso-Britannia: Packt Publishing Ltd.

Wikipedia. 2013a. Skeletal Animation. Viitattu 4.9.2013.  
[http://en.wikipedia.org/wiki/Skeletal\\_animation](http://en.wikipedia.org/wiki/Skeletal_animation)

Wikipedia. 2013b. Skeletal Animation. Viitattu 4.9.2013.  
<http://en.wikipedia.org/wiki/Kinematics>

Normet. 2013. Scamec 2000 S Technical data sheet. Viitattu 9.9.2013  
[http://www.normet.com/inet/normet/flow.nsf/%28AttachmentsByFileName%29/Scamec%202000%20S%20100043782.pdf/\\$File/Scamec%202000%20S%20100043782.pdf](http://www.normet.com/inet/normet/flow.nsf/%28AttachmentsByFileName%29/Scamec%202000%20S%20100043782.pdf/$File/Scamec%202000%20S%20100043782.pdf)

Liitteet merkitään omalla tyylillään:

Liite 1

## LIITTEEN OTSIKKO

Liite-sivulle ei tule alareunan sivunumerointia.