DIGITALE MEDIENPRODUKTION PRODUCTION JANUAR | FEBRUAR 01:2017

CG-Characters!

Im Fokus: Helden-Design und Statisten-Creation

Maya Bifrost

Neue Wassersimulation – bis zum letzten Schluck Neue Tools ZBrushCore, Trapcode, Blender 2.78 & mehr





Glänzend wie Stahl, samtig wie Wachs – oder umgekehrt?

Verblüffende Shader-Setups für Cinema 4D, nur mit Onboard-Tools erstellt: Smart AO und Shadow Luminance simulieren diffusen Schatten und diffuses Licht, Smart Bleed sorgt für Farbbluten in Lichtbereichen und Fast Jelly simuliert mithilfe von Xpresso komplexes Subsurface Scattering.

Per Mensch ist ein Gewohnheitstier. Sehgewohnheiten bestimmen als kulturell beeinflusste Wahrnehmungskonvention, ob wir eine Farbharmonie spannend oder uninteressant finden oder ob die Schnittfolge eines Films stimmig oder eher irritierend wirkt. Aber auch beim Assoziieren von Dingen mit ihren Eigenschaften gelten Gewohnheiten. So belegen wir in unserer täglichen Wahrnehmung bestimmte Materialien mit typischen Attributen. Nehmen wir uns einen Augenblick und visualisieren vor

unserem inneren Auge einmal zwei exemplarische Beispiele: Stahl und Wachs.

Mit der Vorstellung von kaltem, harten Stahl werden wohl als erstes die reflektiven Attribute assoziiert: mehr oder weniger unscharfe Spiegelungen und Glanzlichter, eventuell durch Mikro-Facettierung anisotropisch gestreckt – glänzend wie Stahl.

Wachs wiederum verbinden wir mit komplett anderen Eigenschaften: warm und verformbar, lichtdurchströmt und an seiner Oberfläche satiniert – samtig wie Wachs. Assoziationsgewohnheiten leisten also einen guten Beitrag, um sich in der visuellen und haptischen Welt zurechtzufinden. Doch wie wäre es, wenn man Materialattribute einfach austauschen würde: Samtig wie Stahl, glänzend wie Wachs? Sandgestrahlter Stahl oder eloxiertes Aluminium verbinden wir ebenso legitim mit "samtig" wie die günstige Kerze vom Discounter mit "glänzend".

Gerade das Brechen von Seh- oder Assoziationsgewohnheiten führt in Film- und Bildgestaltung immer wieder zu ungewöhn-



lichen und originellen Ergebnissen. Was spricht also dagegen, sich auch bei der Erstellung von Materialeigenschaften in Cinema 4D dem angestrebten Ergebnis von einer völlig unerwarteten Seite zu nähern?

Im Folgenden betrachten wir ein paar Shader-Setups, die ich über die letzten Jahre entwickelt habe und die mir in meiner täglichen Arbeit in Sachen Shading, Lighting und Rendering eine enorme Hilfe sind. Sämtliche Shader-Setups können anhand der Beispiel-Datei unter www.renderbaron.de/ downloads/renderbaron_smart-shader.zip (dem Drachen auf dem Podest aus dem Titelbild) nachvollzogen werden.

Smart AO – Ambient Occlusion nur in Schattenbereichen

Ambient Occlusion (AO) ist ein in der Computer-Grafik populäres Shading-Verfahren welches einen diffusen Abdunklungseffekt in Vertiefungen und Ecken erzeugt, indem es die Abdeckung von Objekten innerhalb bestimmbarer Distanzen sampelt. Entwickelt wurde das Verfahren 2002 von VFX-Supervisor Hayden Landis und Kollegen bei Industrial Light & Magic.

In Cinema 4D liegt AO als globaler Rendereffekt in den Rendervoreinstellungen und als Channel-Shader vor. Abseits von Clay-Renderings sollte man AO tunlichst als Shader verwenden. Dies ermöglicht deutlich differenziertere und auch effizientere Ergebnisse, da AO je nach Bedarf auf einzelne Objekte angewendet oder ausgelassen werden kann. Auch erfordern kleinere Objekte geringere Sample-Distanzen als große. AO ist insofern nicht physikalisch korrekt, als es einen schattenähnlichen Abdunklungseffekt vollkommen unabhängig von Licht erzeugt. Ambient Occlusion ist ohne weiteres Zutun also auch in direkt beleuchteten Bereichen sichtbar und führt so erst einmal zu einem schmutzigen Erscheinungsbild. Nicht umsonst wurden Vorgänger-Versionen von AO in Cinema 4D als "Dirt-Shader" betitelt. Beobachten wir jedoch das Pendant zu AO in der Realität, nämlich diffuse Schatten, so sind diese in direkt beleuchteten Bereichen prinzipbedingt natürlich nicht vorhanden (Abb. 01).

Mit etwas Know-How und ein paar Handgriffen in Cinema 4Ds flexiblem Materialsystem kann man den AO-Shader allerdings zu einem physikalisch korrekteren Erscheinungsbild verhelfen, getreu dem Motto: "Es muss nicht physikalisch korrekt sein, es muss nur physikalisch korrekt aussehen". Machen wir aus "AO" also "Smart AO".

Klassischer Einsatzort dafür ist der Diffusionskanal eines Cinema-4D-Materials, da dieser Abdunklungen aus den Helligkeitsinformationen seines Inputs (Shader oder Bitmaps) generiert. Ziel ist es also, Ambient Occlusion auf weniger oder nicht beleuchtete Bereiche zu maskieren. Zwei Komponenten werden allein schon durch diese Formulierung klar: Wir benötigen einen Shader, der lichtsensitiv ist und feststellen kann, wo Licht auftrifft und wo nicht. Als Zweites benötigen wir offensichtlich einen Container zum Verrechnen und damit auch Maskieren von Shadern untereinander. Abb. 01: Diffuse Schatten, wie sie Ambient Occlusion simuliert, sind in der Realität nur in schwach oder indirekt beleuchteten Bereichen sichtbar.

Die erste Komponente, der lichtsensitive Shader, verrät sich fast von selbst durch dessen sinnfällige Namensgebung: der Cinema 4D Effekt-Shader "Lumas". Lumas findet sich im Shader-Drop-down-Menü im Material Manager unter "Effekte" und diente vor der Einführung des Reflektivitätskanals mit Cinema 4D Release 16 (2014) zur Erzeugung von multiplen additiven Glanzlichtern. Lumas' Einstellungen enthalten neben Tabs für 3 Glanzlichter (spekulare Reflexion) und Anisotropie auch den Tab "Shader".

Dieser Tab ist im Grunde falsch benannt, denn in Korrespondenz zu den Spekular-Tabs müsste er konsequenterweise "Diffus" heißen, da er sich ausschließlich um die diffuse Beleuchtungsinformation kümmert. Stellt man das dortige Farbfeld auf reines Weiß, den Parameter "Beleuchtung" auf 100% und deaktiviert alle Glanzlichtparameter, funktioniert der Lumas-Shader nicht anders als der Farbkanal – er sagt: "Zeig mich, wo ich im Licht bin." (Abb. 02)

Benennen wir einen so konfigurierten Lumas-Shader im Folgenden der Einfachheit halber als Diffus-only-Lumas.

Die zweite Komponente, der Container zum Verrechnen und Maskieren von Shadern untereinander, findet sich ebenfalls im Shader-Drop-down-Menü und nennt sich "Ebene". In diesem kann man nach Photoshop-Manier Shader und Bitmaps als Ebenen übereinander stapeln, in der Deckkraft INTERACTIVE

	Material-Editor							

	Basis	Shader	Glanzlicht 1	Glanzlicht 2	Glanzlicht 3	Anisotropisch		
Padast	Shader-Eigen	schaften						
rouest	Aktiv	. 🗸						
	O Farbe							
	Algorithmus	Intern						
	O Beleuchtung	100 %						
	Kontrast	. 0%						
O Umgebung	Glanzlicht 1							
Nebel	Aktiv							
🔘 Relief 🖌		•						
🔘 Normale 🔲								
O Alpha ■								
🔘 Glühen 🔲								
Displacement								
Editor								
Illumination	Glanzlicht 2							
Zuweisen	Aktiv							

Abb. 02: Der Lumas-Shader in Diffus-only-Konfiguration: "Zeig mich, wo ich im Licht bin."







Abb. 03: Der Ebene-Shader als Container zur Verrechnung von Shadern. Im Bild ein einfaches Beispiel zur Maskierung eines Shaders durch einen darunterliegenden.

$\Theta \ O$	Material-Editor								
Polor	Basis Shader								
Podest	Name Smart AO								
● Farbe ✔	Ebene								
Diffusion V	Blur-Offset 0 %	+							
🖲 Leuchten 🗸	Blur-Stärke 0 %	+							
🔘 Transparenz 🛛	Shader-Figenschaften								
Reflektivität	Ehenen								
🖲 Umgebung 🔲	Bild Shader	Bild Shader Effekt Ordner Löschen							
Nebel	Ambient Occlusion	Normal							
🔘 Relief 🖌	Ambient Occlusion	Normai	× 001	Ŧ					
🔘 Normale 🔲	Colorizer	Ebenenmaske	✓ 100 %	+					
🔘 Alpha 🔲	Tarbe	Normal	- 100 %	+					
🖲 Glühen 🔲									
Displacement									
Editor									
Illumination									
Zuweisen									

Abb. 05: Das Smart AO Setup: Ambient Occlusion wird durch einen per Colorizer umgekehrten Diffus-only-Lumas auf Schattenseiten des Objekts maskiert. Dadurch leer bleibende Stellen werden durch einen weißen Farbe-Shader grundiert.

variieren und mit unterschiedlichen Ebenen-Modi und Ebenen-Effekten miteinander verrechnen. Zu den Ebenen-Modi gehört neben Altbekannten wie "Normal", "Mulitplizieren" und "Ineinanderkopieren" auch der Modus "Ebenenmaske". In diesem Ebenen-Modus wird die darüberliegende Ebene (also ein Shader oder Bitmap) mit den Graustufen der "Ebenenmaske"-Ebene maskiert (Abb. 03).

Legen wir also mit unserem "Smart AO"-Setup los und erzeugen im Diffusionskanal unseres Materials durch Klick auf den Dreieck-Knopf neben "Textur" zunächst einen Ebenen-Shader als Container. Im Ebenen-Shader erzeugen wir durch Klick auf "Sha-

Abb. 06: Indirektes Licht in unterschiedlich verortbarer Form: der Untergrund mit sehr diffusem Licht in Schattenbereichen und der Teller mit Colorbleeding in den direkten Kontaktbereichen zu den Früchten.





Abb. 07: Das Shadow Luminance Setup: Farbinformation wird durch einen per Colorizer umgekehrten Diffus-only-Lumas maskiert und erzeugt so ein leichtes Selbstleuchten auf Schattenseiten des Objekts.

der" einen Ambient-Occlusion-Shader und justieren dort je nach Beschaffenheit der Szene nur die "Maximum Strahlenlänge", um zu definieren, wie weit AO seine Umgebung abtastet. (Abb. 04)

Zurück im Ebenen-Shader erzeugen wir einen Lumas-Shader in "Diffus-only"-Konfiguration (siehe oben). Danach ziehen wir in unserem Ebenen-Shader den Lumas-Shader unter den AO-Shader und stellen den Ebenen-Modus des Lumas auf "Ebenen-Maske". Als Resultat maskieren wir so AO auf direkt beleuchtete Bereiche.

Nun müssen wir einen Weg finden, um das Verhalten des Lumas "Zeig mich, wo ich im Licht bin" umzukehren in "Zeig mich, wo ich im Schatten bin". Für das Uminterpretieren und damit auch Umkehren von Shadern gibt es im Shader-Set von Cinema 4D einen idealen Kandidaten: den Colorizer-Shader. Dieser interpretiert die Graustufen-Informationen seines Inputs (Shader und Bitmaps) in einen frei definierbaren Farbverlauf.

In unserem Ebenen-Shader rechtsklicken wir auf unseren Lumas-Shader und wählen "Colorizer". Dadurch werfen wir den bestehenden Lumas quasi als Input in einen neuen Colorizer-Shader. Im Colorizer-Shader ändern wir den schwarz-rot-gelb-Verlauf auf einen schwarz-weiß-Verlauf. Die Graustufen-Informationen des beherbergten Lumas-Shaders werden nun 1:1 original interpretiert: schwarz ist schwarz und weiß ist weiß. Führen wir jedoch einen Rechtsklick in den Colorizer-Verlauf aus und wählen "Knoten umkehren", so invertieren wir diese Interpretation und damit die Wirkung des Lumas-Shaders: aus "Zeig mich, wo ich im Licht bin" wird "Zeig mich, wo ich im Schatten bin". Der AO-Shader wird also auf die Schattenseite von Objekten maskiert. Allerdings werden durch die Maskierung leerbleibende Bereiche vom Diffusionskanal als "schwarz" interpretiert. Um das zu beheben, erzeugen wir im Ebenen-Shader mit Klick auf "Shader" einen Farbe-Shader, der voreingestellt rein weiß ist. Diesen Farbe-Shader ziehen wir als Grundierung unterhalb des Colorizer-Shaders et voila, unsere Testszene zeigt: Ambient Occlusion wird ohne Artefakte auf Schattenbereiche maskiert. Nachträglich ist es

nun noch sinnvoll, den Kontrast-Parameter des AO-Shaders auf 20 bis 30% anzuheben, um den nun maskierten AO-Effekt etwas zu verstärken. (Abb. 05) Im Basis-Tab unseres Ebenen-Shaders können wir unser Setup nun noch als "Smart AO" benennen. Zurück im Diffusionskanal klicken wir noch auf den Dreieck-Knopf neben "Textur", wählen "Preset speichern" und haben Smart AO so dauerhaft als Preset verfügbar.

Shadow Luminance

Trifft Strahlung gleich welchen Spektrums auf Oberflächen, wird diese Strahlung mehr oder weniger reflektiert, abgelenkt und gestreut. Im Bereich des sichtbaren Lichts spricht man dabei von direktem und indirektem Licht. Ohne den Einsatz von Automatismen, wie z.B. der Berechnung globaler Illumination (GI), erfordert die Erzeugung indirekten Lichts in einer 3D-Szene eine hohe Qualifikation des Artists. Dabei trägt im Prozess des Lightings gerade indirektes Licht massiv zur Glaubwürdigkeit einer Szene bei.

Indirektes Licht kann dabei in mehr oder wenigerverortbaren Varianten auftreten: zum Beispiel als sogenanntes "Farbbluten" (Colorbleeding), welches eindeutig einem direkt beleuchteten farbigen Objekt als Quelle zuzuordnen ist oder aber als diffuse Illumination auf nicht direkt beleuchteten Seiten eines



Abb. 08: Interior-Szene mit manueller Ausleuchtung (ohne GI); obere Abbildung ohne und untere mit Einsatz von Shadow Luminance. Besonders Sofa-Decke und Raum-Ecken profitieren.

FILM & VFX

3D & ANIMATION INT

INTERACTIVE DIGITAL ART

SCIENCE & EDUCATION





Abb. 09: Unterwasserszene für ZDF Terra X: Hunderte von Korallen mit Subsurface Scattering treiben die 64 GB RAM der Render-Clients an ihre Grenzen.

Objekts. (Abb. 06) Versuchen wir einmal einen unkonventionellen Denkansatz: Könnte man letzteren Fall – sehr diffuses indirektes Licht in Schattenbereichen – nicht auch als Selbstleuchten durch den Leuchten-Kanal eines Cinema-4D-Materials erzeugen?

Dazu müsste die Primärfarbe eines Objekts als Selbstleuchten auf seine Schattenseiten maskiert werden. Eine Abwandlung des Smart-AO-Shader-Setups kann dazu verwendet werden, genau das zu bewerkstelligen. Im Material unseres Objekts rufen wir das oben erzeugte Smart-AO-Preset auf - diesmal jedoch im Leuchten-Kanal. Im Ebenen-Shader entfernen wir zunächst den AO- und den weißen Farbe-Shader. Nun kopieren wir den Inhalt des Farbe-Kanals in unseren Ebenen-Shader und fügen ihn oberhalb des Colorizer-Shaders ein (Shader-Drop-down-Menü "Shader kopieren", "Shader einfügen"). Alternativ erzeugen wir einen Farbe-Shader in der ungefähren Primär-Farbe des Inhalts des Farbkanals. Die über dem Colorizer-Shader liegende Ebene wird nun auf die Schattenseite des Objekts maskiert. Nun müssen wir nur noch die Deckkraft dieser Ebene herunterregeln. 5 bis 10% sind hierbei ein guter Startwert. (Abb. 07) Unter "Basis" benennen wir unser Setup als "Shadow Luminance" und speichern es ebenfalls als Preset für weiteren Gebrauch.

Shadow Luminance ist ein hervorragender, auf Shadern basierender Ansatz, um die ansonsten aufwendige Erzeugung sehr diffusen indirekten Lichts zu simulieren, bzw. zu ergänzen. Es kann in vielen Fällen durchaus die manuelle Platzierung von extra Lichtquellen oder gar den Einsatz von Global Illumination ersetzen (Abb. 8), besonders im Wechselspiel mit Smart AO, wie wir nun sehen werden.

Shadow Luminance im Wechselspiel mit Smart AO

Hat man Smart AO im Diffusionskanal und Shadow Luminance im Leuchten-Kanal geladen, können beide miteinander interagieren, indem man für Smart AO im Diffusionskanal die Checkbox "Wirkung auf Leuchten" aktiviert. Dadurch wird definiert, dass Smart AO in seinem Wirkungsbereich den Effekt von Shadow Luminance aufhebt. Ohne das Aktivieren dieser Option würde Shadow Luminance die Wirkung von Smart AO überschreiben. Mit Smart AO und Shadow Luminance haben wir überraschende Shader-Setups kennengelernt, die bei der Simulation von diffusen Schatten und indirektem Licht mächtige Verbündete sein können – und zwar auf Shadern basierend, effizient und interaktiv. Die Setups sind durch das Speichern als Presets dabei jederzeit zur Hand und können mit wenigen Klicks angepasst werden.

Smart Bleed

Die direktere Variante indirekten Lichts, das eingangs erwähnte Colorbleeding (Abb. 6), kann in Einzelfällen durch ein ähnliches Setup erstellt werden. Dessen Prinzip ist es, im Leuchten-Kanal bei Annäherung eines Objekts einen farbigen AO-Effekt auf die Lichtseiten eines Objekts zu zaubern. Dazu erzeugt man im Leuchten-Kanal einen Ebenen-Shader und benennt diesen als "Smart Bleed". Im Ebenen-Shader erstellt man einen AO-Shader mit einem Farbig-Schwarz-Verlauf. Als Farbe wird die primäre Farbe des farbblutenden Objekts gewählt. Schwarz wird im Leuchtenkanal als inaktiv interpretiert. Zurück im Ebenen- Shader erzeugt man einen Lumas-Shader in Diffus-only-Konfiguration (siehe oben) und maskiert den AO-Shader durch vorhin erklärtes Vorgehen auf die Lichtseiten des Objekts. Dann speichert man das Ganze als Preset. Wir haben nun einen Shader erstellt, der nur im direkten Licht einen Farbbluten-Effekt passend zur Farbe des Objekts erzeugt - in unserem Fall einen leicht rosafarbenen Schein auf der Standfläche des Podests. (siehe Titelbild)

Fast Jelly

Mit etwas weiterem Um-die-Ecke-Denken lassen sich allerdings nicht nur Licht- und Schatteneffekte auf der Oberfläche von Objekten, sondern auch in deren Volumen simulieren, sprich: Subsurface Scattering (SSS). SSS beschreibt die Lichtstreuung im Volumen transluzenter Materialien, wie zum Beispiel Marmor, Wachs oder Haut. Die Shading-Methode wurde 1998 erstmals von Henrik Wann Jensen am MIT ausgearbeitet und ist in Cinema 4D als einfach zu bedienender und verhältnismäßig schnell zu berechnender Effekt-Shader verfügbar. Warum also in Eigenbau einen Ersatz dafür erstellen? Bei der massenhaften Anwendung von SSS, zum Beispiel bei der Darstellung zahlreicher Pflanzen, wird schnell klar, wie sehr dieser Shader nach RAM hungert. So ist es in einem unserer letzten Projekte für das ZDF dazu gekommen, dass für die Darstellung eines Korallenriffs mit SSS die 64 GB RAM eines jeden Render-Clients gerade einmal so ausreichten. (Abb. 09) Die RAM-Knappheit herrschte dabei unabhängig davon, ob SSS im Cache- oder Direkt-Modus angewendet wurde. Dies veranlasste mich, nach einer Alternativ-Lösung zur Darstellung SSS-ähnlicher Effekte zu suchen sei es f
ür den Produktionseinsatz oder nur als sexy Proof-of-Concept. Analysieren wir, welche Merkmale SSS in seiner Erscheinung auszeichnen, können wir drei Feststellungen treffen: 1) SSS kann nur mit einem volumenwirksamen Shader dargestellt werden. 2) SSS ist vor allem am Übergang zwischen Licht und Schatten, bzw. auf der Schattenseite von Objekten sichtbar. 3) SSS ist zur Oberfläche bzw. zur Hauptlichtquelle hin heller.

Bei der Implementierung als Shader-Setup für den Drachen der Beispieldatei erscheinen die ersten zwei Bedingungen als leicht umsetzbar: als volumenwirksame Shader kennen wir die in Cinema 4D enthaltenen 3D-Farbverlauf-Shader. Die Maskierung von Shadern auf die Schattenseite von Objekten beherrschen wir mittlerweile auch. Das bedeutet, unser Fast-Jelly-Setup wird auf einem Ebenen-Shader im Leuchten-Kanal beruhen, bei dem ein 3D-Farbverlauf auf die Schattenseite des Objekts maskiert wird. Nur die automatische Ausrichtung des 3D-Farbverlaufs auf die Hauptlichtquelle scheint noch rätselhaft.

Sehen wir uns für eine Lösung einmal die Parameter eines solchen 3D-Verlaufs an. Erzeugen wir im Leuchten-Kanal zunächst einen Ebenen-Shader als Container und darin einen Farbverlauf-Shader im Modus "3D Linear". Die Ausrichtung des 3D-Farbverlaufs wird entlang der drei Raumachsen X, Y und Z mittels eines Start- und Endpunktes definiert. So lässt sich der Farbverlauf in alle Himmelsrichtungen drehen. Eine Ausrichtung des Verlaufs auf eine Lichtquelle ist dabei jedoch kaum intuitiv, geschweige denn interaktiv, falls sich die Lichtquelle bewegt.

Beim Stichwort "ausrichten" fällt uns allerdings ein Cinema-4D-Tag ein, welches die Z-Achse eines Objekts auf beliebige Zielobjekte ausrichtet: das Ausrichten-Tag. Als Lösungsansatz für die Ausrichtung unseres 3D-Verlaufs wäre also folgendes denkbar: ein Referenz-Objekt an der ungefähren Position unseres Drachen wird per Ausrichten-Tag auf die Hauptlichtquelle ausgerichet und überträgt seine Ausrichtung auf diejenige des 3D-Farbverlaufs. Vorsicht: Rocket Science!

Fast Jelly plus Xpresso

Das klingt ganz nach Xpresso, Cinema 4Ds Node-basiertem Editor für ObjektinterakSHADER I CINEMA 4D



Abb. 10: Fast Jelly mithilfe von Xpresso: die Polpunkte einer auf das Hauptlicht ausgerichteten Kugel werden per Punkt-Nodes an Start- und Endpunkt eines 3D-Farbverlaufs übergeben. Links der X-Pool, aus dem die Nodes gewählt werden.

$\Theta \Theta \Theta$		Material-Editor						
***							-	
	Basis Shader							
Drache	Name Fastlelly							
- · 🗞	Fbene							
Farbe								
Diffusion V	Blur-Offset 0% +							
O Leuchten V	Blur-Stärke 0 % +							
Transparenz	Shader-Eigenschaften							
Reflektivität	Ebenen							
Umgebung	Bild Shader Effekt O	rdner Löschen						
Relief	Vertiefungen/ Ambient Occlusion	Multiplizieren		100 %	+			
Normale	Inverse Ambient Occlusion	Ineinanderkopieren		30 %	+			
O Alpha 🔲	Colorizer	Ebenenmaske		100 %	+			
🔘 Glühen 🔲	- Drdner	Normal		100 %				
🔘 Displacement 🔲		Korma		100 %	*	-	_	
Editor	Fresnel	Ineinanderkopieren		20 %	+			
Illumination	3D-Farbverlauf	Normal		100 %	+	1		
Zuweisen	Colorizer	Ebenenmaske	-	100 %	+			

Abb. 11: Das fertige Fast-Jelly-Setup: ein auf die Hauptlichtquelle ausgerichteter 3D-Verlauf, maskiert auf die Schattenseite und überlagert von drei weiteren Effekt-Shadern.

tionen. Erzeugen wir also ein solches Referenz-Objekt, eine Kugel, an der Position und in ungefährer Größe unseres Drachen. Wandeln wir danach die Kugel mit "C" in ein Polygon-Objekt um und drehen das Achsen-System so um 90 Grad, dass die Z-Achse durch die Pole führt. Folgend wird die Kugel mit einem Ausrichten-Tag auf die Hauptlichtquelle in unserer Szene ausgerichtet. Ergebnis: Nord- und Südpolpunkt der Polygon-Kugel zeigen interaktiv auf die Hauptlichtquelle. Wir werden nun die Position des Nord- und Südpolpunktes der Kugel auf Start- und Endpunkt unseres 3D-Farbverlaufs übertragen.

Im Struktur-Manager von Cinema 4D ermitteln wir dazu, welche Punkte-Nummer diese beiden Punkte haben. Dann erstellen wir für die Kugel ein Xpresso-Tag, doppelklicken es und erzeugen im Xpresso-Editor zunächst ein Objekt-Node, auf das wir unsere Kugel ziehen. So definieren wir, von welchem Objekt der ganze Zauber überhaupt handelt.Danach werden zwei Punkt-Nodes erzeugt. (Die nun beschriebenen Ports müssen jeweils mit einem Rechtsklick

auf die rote oder blaue Ecke eines Nodes neu erstellt werden.) Den Objekt-Port des Objekt-Nodes verbinden wir dann mit dem jeweiligen Objekt-Port der beiden Punkt-Nodes. Im jeweiligen Attribut-Manager der Punkt-Nodes geben wir unter "Punkt-Index" die Zahl ein, die wir im Struktur-Manager für Nord- und Südpol-Punkt der Kugel ermittelt haben.

Bei den Punkt-Node-Ports "Punkt-Position" handelt es sich um Vektoren mit 3 enthaltenen Koordinaten für X, Y und Z genauso wie Start- und Endpunkt unseres 3D-Verlaufes. Wir müssen also nur noch die Punkt-Position-Ports mit Start- bzw. Endpunkt-Ports des 3D-Verlaufs verbinden. Um den 3D-Verlauf als Node in den Xpresso-Editor zu bekommen, klicken wir in unseren 3D-Verlauf-Shader und ziehen das Vorschaubildchen des Verlaufs in den Xpresso-Editor. Die genannten Ports verbinden – fertig. (Abb. 10)

Bei der Abarbeitung des Ausrichten- und des Xpresso-Tags durch Cinema 4D ist es wichtig, dass das Ausrichten-Tag zuerst berücksichtigt wird, um dann erst die Position

FOKUS

FILM & VFX

3D & ANIMATION

INTERACTIVE DIGITAL ART







der beiden relevanten Kugel-Punkte in das Xpresso-Prozedere zu übergeben.

Es ist daher sinnvoll dem Ausrichten-Tag im Attribut-Manager eine höhere Priorität (zum Beispiel "Expression 10") zu geben als dem Xpresso Tag (zum Beispiel "Expression O"). Wir haben nun den 3D-Farbverlauf unseres Fast-Jelly-Shaders über ein Polygon-Objekt mit Ausrichten-Tag interaktiv auf die Hauptlichtquelle ausgerichtet: bewegen wir die Lichtquelle, dreht sich die Kugel mit und damit der per Xpresso verlinkte 3D-Verlauf in unserem Fast-Jelly-Setup. Diese interaktive Ausrichtung ist auch in dem in den Beispieldateien enthaltenen Animations-Clip gut zu sehen. Damit haben wir die Basis für Fast Jelly erstellt. Im fertigen Setup ist der 3D-Farbverlauf nun mittels eines per Colorizer umgekehrten Diffus-only-Lumas (siehe oben) auf die Schattenseite des Drachen maskiert. Zur Verfeinerung des Looks liegen weitere Shader über dem so maskierten 3D-Verlauf. Abb. 12: Shading-Know-how auf eigenen Wegen: selbst erstellter Holz-Shader mit Kantenbeschädigungen, Atmosphäre mit volumetrischen Wolken und voll prozedurales Landscaping.

So sorgt ein Fresnel-Shader für eine satinierte Anmutung. Darüber liegt ein AO-Shader, der durch die in R18 neue Option "Gegenrichtung" ausschließlich Kanten und dünne Stellen zum Leuchten bringt. Dieser Effekt ist an sich schon eine einfache Simulation von Subsurface Scattering. Da er jedoch nicht wie ein 3D-Verlauf das Volumen des Drachen durchdringt, wird er hier nur als schwach deckender Zusatzeffekt auf Schattenseiten verwendet. Außerdem sorgt ein multiplizierter rot-weißer Ambient-Occlusion-Shader für rötliche Abdunklungen in Vertiefungen. (Abb. 11)

Fazit

Mit Smart AO und Shadow Luminance haben wir Shader-Setups kennengelernt, die den täglichen Workflow bei Shading, Lighting und Rendering massiv erleichtern. Smart Bleed kann in Einzelfällen das Erzeugen von Colorbleeding ersetzen, und Fast Jelly zeigt, wie Xpresso den Shading-Prozess erweitern kann. "Es muss nicht physikalisch korrekt sein, es muss nur physikalisch korrekt aussehen" physikalisch korrektes Shading steht heute als Resultat großartiger Forschungsarbeiten beinahe jedem Artist in Form moderner 3D-Software zu Verfügung. Dennoch sollte diese Verfügbarkeit nicht zu einem formalen technischen Akademismus verleiten, der Abweichungen und Neuerungen auf abseitigen Wegen nicht zulässt. Gerade das Entwickeln eigener Shading- und auch Lighting-Lösungen außerhalb der technischen Tradition ist ein hervorragendes Mittel, den eigenen Horizont zu öffnen und nicht einem Fanboyismus zu verfallen. Die geschilderten Shader-Setups für Cinema 4D sind dabei nur ein Ausschnitt der Machbarkeiten. Diese reichen von eigens erstellten Oberflächen-Shadern für Holz, Fels und Wasser über volumetrische Atmosphären- und Wolken-Systeme bis hin zu vollprozeduralem Landscaping vom Close-up bis zum planetaren Maßstab. (Abb. 12) "Think outside the Box" ist zum Erforschen dieser Möglichkeiten dabei genau der richtige Ansətz. >ei



Marc Potocnik ist Diplom-Designer (FH) und Gründer/Inhaber des Animationsstudios renderbaron in Düsseldorf. renderbaron realisiert seit 2001 hochwertige 3D-Animationen und Visual Effects für renommierte Kunden wie ZDF, Audi, BMW, Siemens, und andere. Marc Potocnik teilt sein Wissen als Maxon Certified Instructor, Schwerpunkt "Shading, Lighting & Rendering" und als gefragter Sprecher auf internationalen Fachkonferenzen wie der Siggraph, FMX oder IBC.

www.renderbaron.de www.fb.com/renderbaron

WWW.DIGITALPRODUCTION.COM



ISSN 1433-2620 B 43362 20th Annual Edition Published by ATEC Presented by DIGITAL PRODUCTION

Funded by Bavarian Ministry of Economic Affairs and Media, Energy and Technology



20th JUBILEE Background info, interviews and a look back

MAKING-OF The creative process behind all nominees and award winner

BACKSTAGE Conference highlights, awards gala, animago trailers and more

 Germany
 €
 6,95

 EU/Worldwide
 €
 6,95

 Switzerland
 sfr
 8,59

2016

Get yours at animago.com/shop

CHAO2GROUP





Funded by:



Bayerisches Staatsministerium für Wirtschaft und Medien, Energie und Technologie

畜

Landeshauptstadt München **Kulturreferat**











PNY

VOGELSÄNGER

