

2019

ISSN 1433-2620 > B 43362 >> 23. Jahrgang >>> www.digitalproduction.com

Publiziert von DETAIL Business Information GmbH

Deutschland € 17,90
Österreich € 19,-
Schweiz sfr 23,-

3

DIGITAL
PRODUCTION

DIGITAL PRODUCTION

MAGAZIN FÜR DIGITALE MEDIENPRODUKTION

MAI | JUNI 03:2019



Ausbildung!

Die besten Schulen und Projekte der Republik

Praxis!

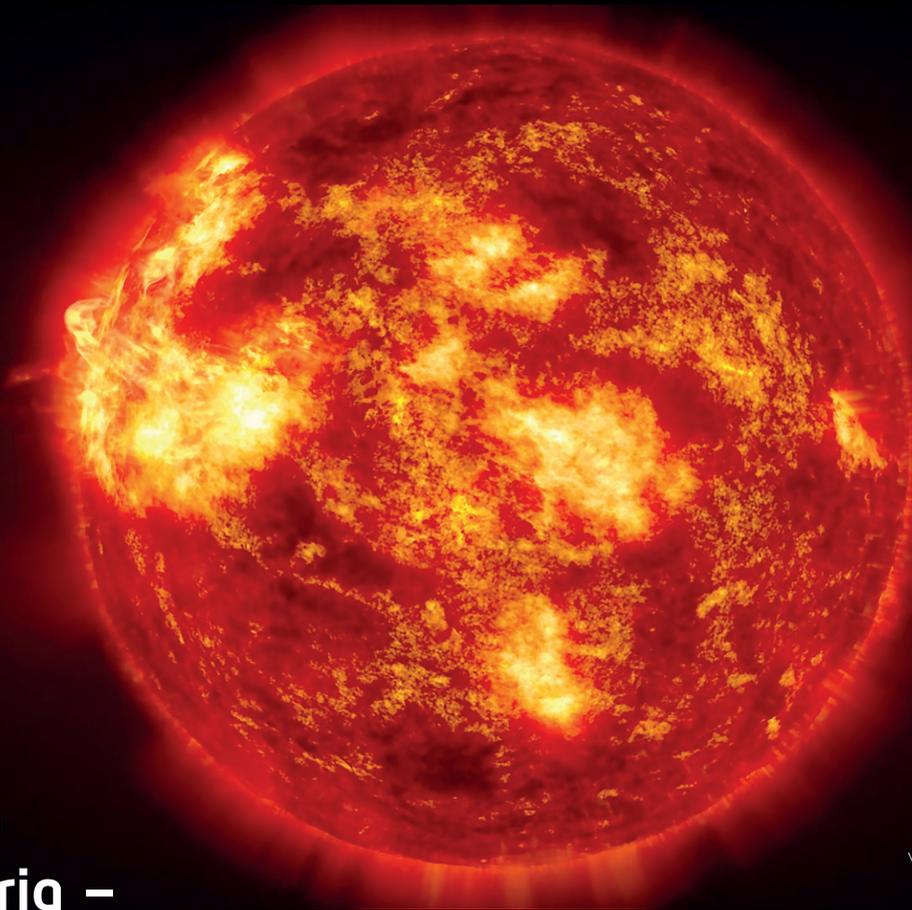
Houdini 17.5, Blender 2.8, Lightwave, Rayfire & C4D

und vieles mehr!

Fractures, 3D LC, Andrew Kramer, Titan RTX, ...



Sonne aus ZDF „Terra X“, Episode „Eine Frage der Zeit“ (<https://vimeo.com/renderbaron/txeine-fragederzeit>): volumetrische Lichter mit entspr. volumetrischen Texturen erzeugen komplex aussehende Effekte – mit vorhersagbaren Ergebnissen und ohne schlaflose Nächte.



Wolzig bis feurig – Volumetrische Effekte in Cinema 4D

www.renderbaron.de

Teil 2: Sonnen-Noise und Scheibchen-Wolken

In dieser 2-teiligen Artikelreihe eröffnen wir dem gewandten Cinema-4D-Nutzer tiefgehende Einblicke in die Erstellung volumetrischer Effekte mit den Onboard-Tools von C4D – und das ganz ohne Plug-ins, Partikel oder PyroCluster. Im 2. Teil der Reihe vertiefen wir zunächst unsere Kenntnisse zu volumetrischer Texturierung von sichtbaren Lichtern und wenden uns dann einer Mograph-basierten Technik zu.

von Marc Potocnik

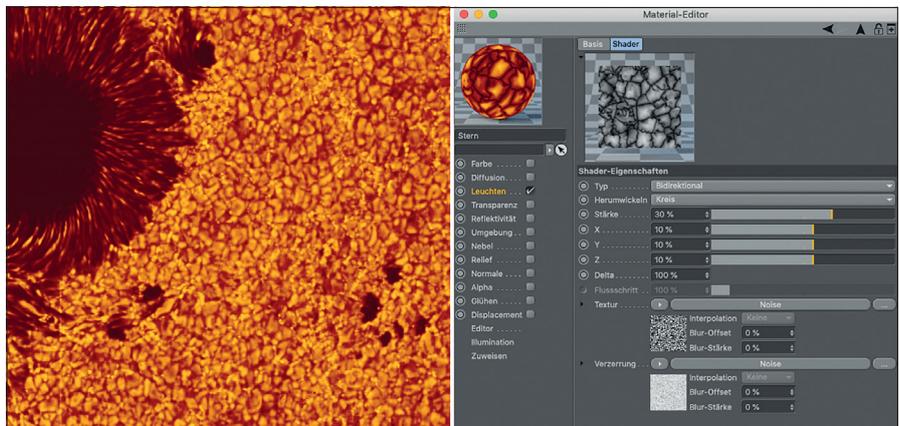


Bild 01: Sonnen-Phänomene und ihre Entsprechungen in C4D: Granule (links) können in C4D leicht erzeugt werden, indem man in einem Distorter Shader ein Voronoi-3-Noise mit einem VL-Noise verzerrt (Foto Granule: NASA, genutzt unter Creative-Common-Lizenz).

Im ersten Teil dieser Artikelserie haben wir gelernt, dass ein sichtbares volumetrisches Licht als „Aquarium“ für volumetrische Shader wie 3D-Noise und 3D-Farbverläufe genutzt werden kann. Mit einem solchen Licht-Container sind wir in der Lage, dünne Wolken oder auch komplexe selbstleuchtende Strukturen wie stellare Nebel zu erzeugen. Mit derselben Technik sind wir jedoch auch in der Lage, komplexe animierte Strukturen zu erzeugen wie z.B. eine animierte Sonnenkorona mit ihren charakteristischen Eruptionen. Im Folgenden werden wir „die Magie sichtbarer Lichter“ also noch eine Stufe weiterführen und uns die Erstellung einer realistischen, feurigen und detailreichen Sonnenkorona vornehmen.

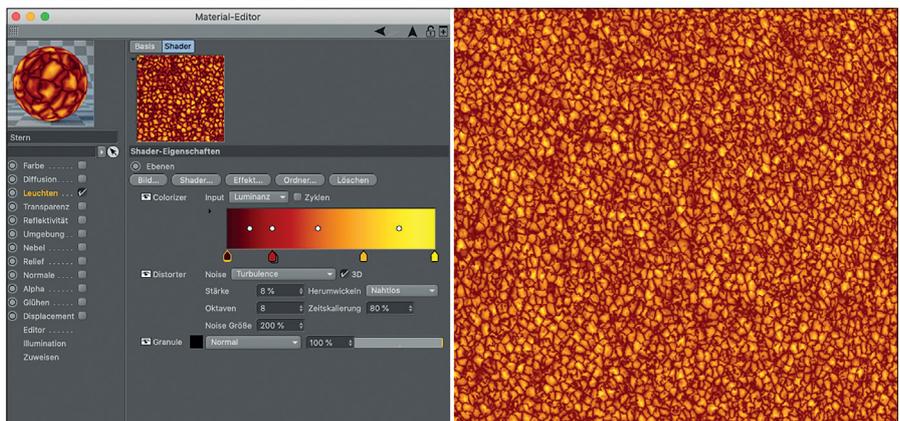


Bild 02: Granule finalisiert: Den fertigen Distorter Shader packt man in einen Ebene-Shader, wendet den Ebenen-Effekt Distorter für eine weitere leichte Verzerrung an und färbt das ganze mit dem Ebenen-Effekt Colorizer.

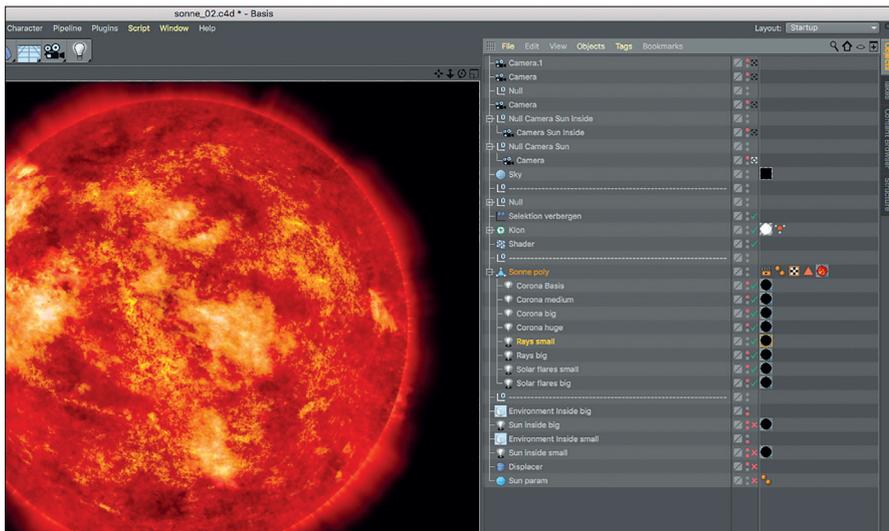


Bild 03: Nur acht volumetrische Punktlichter mit entspr. volumetrischen Texturen mimen komplexe Eruptionen. Einfacher geht's nicht.

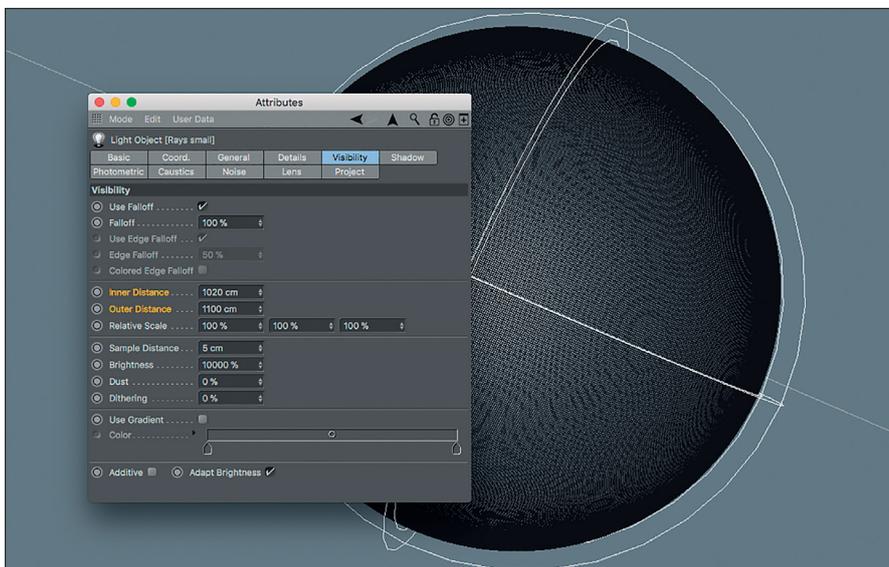


Bild 04: Der Innenradius der Punktlichtquelle entspricht in etwa dem Sonnenradius. Der Außenradius bestimmt die Höhe des darüber hinausgehenden sichtbaren Effekts, also bis wohin Strahlen und Eruptionen über der Sonnenoberfläche sichtbar sein sollen, bzw. auf null abnehmen.

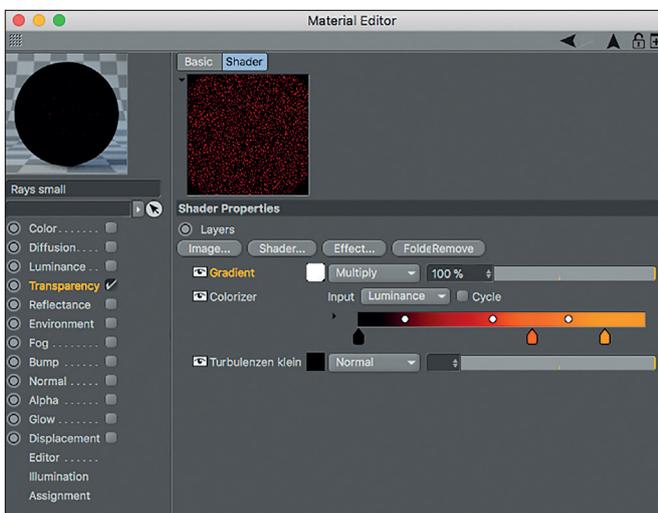


Bild 05: Ein kugelförmiger 3D-Verlauf (Gradient) mit Radius des Außenradius wirkt durch den Ebenen-Modus Multiplizieren quasi maskierend und verfeinert die Abnahme mit einer leichten Turbulenz.

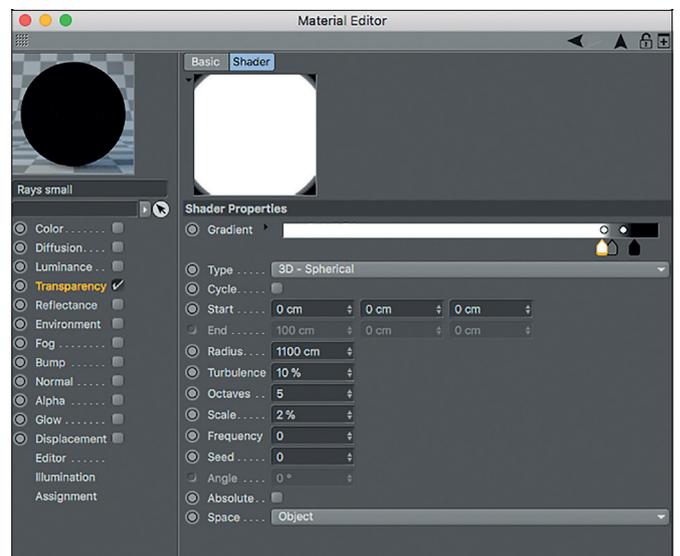


Bild 06: Wichtig ist bei dem kugelförmigen 3D-Verlauf das Bezugssystem Objekt.

Fallstudie: ZDF Terra X, eine Frage der Zeit (Sonne)

Mit Blick auf zeitgenössische Science-Fiction-Filme oder VFX-lastige TV-Dokumentationen könnte man meinen, dass die Darstellung einer realistisch animierten Sonne eine Aufgabe sei, die in einem VFX-Studio als klassisches Szenario für Partikel und Simulationen für viele schlaflose Nächte sorgt.

Die Intro-Sequenz dieses Projekts für die TV-Dokureihe ZDF „Terra X“ (siehe Titelbild) inszeniert die Sonne als Ort stellarer Kernfusion, verwendet jedoch nur die Techniken, die wir bisher kennengelernt haben – und an diesem Punkt können wir wieder beruhigt durchschlafen.

Wenn man sich nämlich einmal NASA-Filmmaterial der Sonne ansieht, wird man feststellen, dass die Sonnenoberfläche jede Menge visuelle Details enthält, die mit C4Ds Noise Shadern vergleichbar sind. So kann beispielsweise die sogenannte Granulation der Sonnen-Photosphäre leicht von einem Noise Shader namens „Voronoi 3“ reproduziert werden, den man in einem Distorter Shader mit einem VL Noise verzerrt (Bild 01).

Wenn man das Ergebnis in einem Ebene-Shader noch mal durch einen Ebene-Effekt „Distorter“ mit einem Turbulence-Noise verzerrt, erhält man eine Struktur, die dem realen Vorbild schon recht nahe kommt (Bild 02).

Das Gleiche gilt für Phänomene, die offensichtlich volumetrisch sind, wie Korona, Sonneneruptionen etc.: sichtbares volumetrisches Licht kann als „Aquarium“ für Noises und 3D-Farbverläufe genutzt werden, die mathematischen Funktionen hinter den Noises ähneln dabei immer wieder ihren natürlichen Vorbildern.

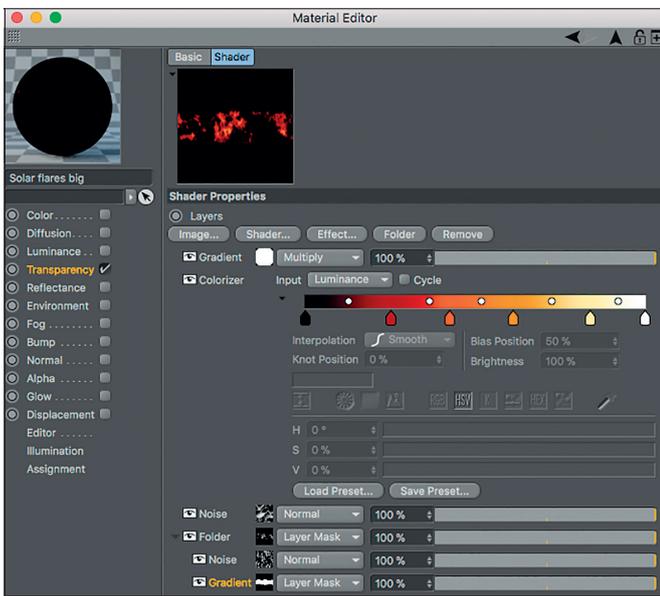


Bild 07: Shader für größere Sonnen-Eruptionen: Zuerst maskiert ein Ebenen-Ordner mit einem 2D-Gradienten das darüberliegende 3D-Noise von den Polkappen der Sonne aus. Oben auf fadet ein kugelförmiger Verlauf das ganze turbulent in den Raum aus.



Bild 09: Die Oberfläche der Sonne: Ein gut gefüllter Ebene-Shader mit sinnfällig benannten Noises, Verläufen, Ebenen-Effekten und Ebenen-Ordner.

Simple Bestandteile

Wie so oft bei Visual Effects ergibt sich die Komplexität der finalen Sonne aus der Überlagerung und Kombination verschiedener an sich einfacher Elemente: Für alle visuellen Phänomene wie Lichtstrahlen unterschiedlicher Größe, Eruptionen usw. verwenden wir jeweils nur ein Punktlicht, dessen sichtbares Licht mehr oder weniger größer als der Sonnenkörper selbst ist. Das Erstaunliche ist: Die Komplexität des Endergebnisses wird nur mit 8 volumetrischen Punktlichtern erreicht (Bild 03).

Konzentrische Maskierung

Korona, Lichtstrahlen und Eruptionen werden also mit der uns bekannten Technik

als Texturen auf volumetrische Punktlichter aufgebracht. Da das Tab „Sichtbarkeit“ des jeweiligen Punktlichts die Parameter „Innenabstand“ und „Außenabstand“ beinhaltet, können volumetrische Effekte konzentrisch von der Sonnenoberfläche in den Raum hinaus ausgeblendet werden (Bild 04).

Um mehr Kontrolle über dieses Verhalten zu haben, wenden wir zusätzlich einen Farbverlauf im Modus 3D-sphärisch als Maske an (Bild 05). Dieser Farbverlauf besitzt denselben Radius wie der Parameter „Außenabstand“ der Punktlichtquelle, hat aber den Vorteil, dass er die räumliche Ausblendung der Effekte feinjustiert und gleichzeitig die Möglichkeit bietet, eine Turbulenz anzuwenden (Bild 06).

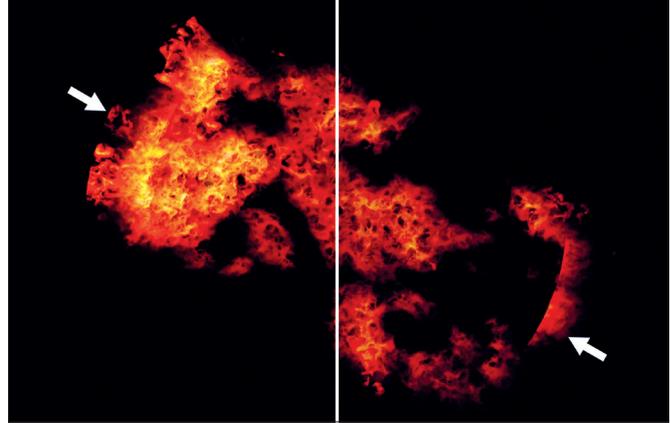


Bild 08: Vergleich der Wirkung des kugelförmigen 3D-Verlaufs: links schneidet die Begrenzung des sichtbaren Lichts die Noise-Textur einfach ab, rechts findet ein turbulentes Ausblenden statt.

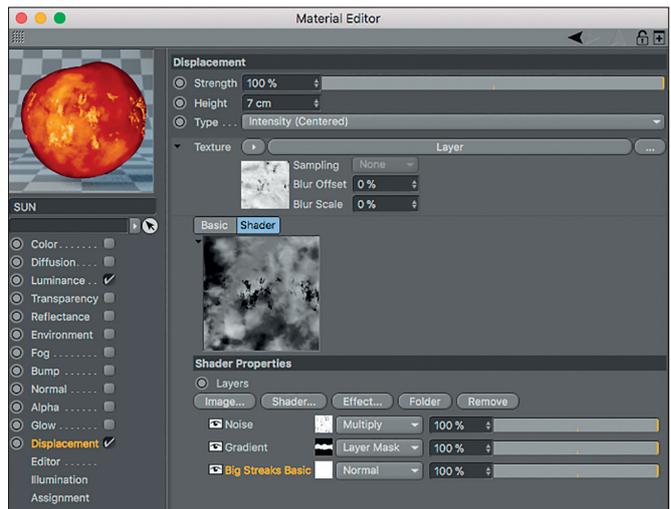


Bild 10: Im Displacement-Kanal sorgen zwei Noises und eine Verlaufsmaske für eine leichte Verformung der Sonne gemäß den Details im Leuchten-Kanal.

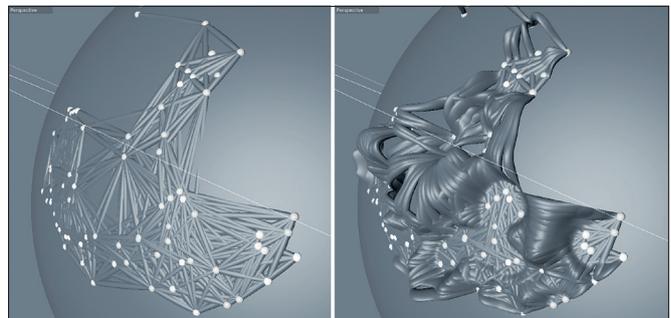


Bild 11: Prozedural modellierte Plasma-Bögen: das Plug-in pConnector schafft Verbindungen zwischen Klonen, ein Displace-Deformer erzeugt die Bogenstrukturen.

Sonneneruptionen

Dieser Effekt mag sehr komplex aussehen, ist tatsächlich aber recht einfach: Er besteht aus einem sichtbaren volumetrischen Punktlicht, das ein großflächiges Naki-Noise im Raum Objekt als Textur enthält. Das Naki enthält durch den Parameter „Animationsgeschwindigkeit“ eine leichte In-sich-Bewegung. Er sitzt in einem Ebene-Shader, wird dort durch einen 2D-Farbverlauf von den Polkappen aus maskiert und durch einen Colorizer-Effekt gefärbt (Bild 07).

Dann wird die Skalierung dieses Naki-Noises über die Zeit hinweg animiert, das heißt: Das Noise wächst konzentrisch vom Zentrum der Sonne ausgehend. Allein da-



© ZDF 2015

www.renderbaron.de



© ZDF 2015

www.renderbaron.de

Bild 12, 13, 14: ZDF „Terra X“-Episode „Mythos der Highlands“ (<https://vimeo.com/renderbaron/txschottland>): fluffige volumetrische Wolken durch Mograph-Kloner als Container für volumetrische Shader

durch entsteht bereits die Illusion plasma-ähnlicher Großstrukturen.

Um mehr Kontrolle darüber zu haben, wie das wachsende Naki im Raum ausgeblendet wird, habe ich es mit einem statischen, aber hoch turbulenten 3D-Farbverlauf im Modus 3D-sphärisch maskiert, der ebenfalls im Raum Objekt angewendet wird (Bild 08). Das Ergebnis ist: Sonneneruptionen entstehen aus der Sonne und werden im Raum turbulent ausgeblendet.

Die Oberfläche der Sonne

Obwohl es in diesem Artikel um volumetrische Effekte geht, werfen wir einen kurzen Blick auf das Shader Setup für die tatsächliche Oberfläche der Sonne. Alles spielt sich im Leuchten-Kanal des Sonnen-Materials ab bzw. in einem dort platzierten Ebene-Shader: Noises, Farbverläufe und Ebene-Shader-Effekte werden mit verschiedenen Ebenenmodi und Deckungsgraden aufeinander gestapelt. Außerdem werden mehrere Ebenen verwendet, um die darüberliegende zu maskieren (Bild 09).

Im Displacement-Kanal verformen drei Noises in einem Ebene-Shader die High-Res-Geometrie der Sonne passend zu den Details im Leuchten-Kanal (Bild 10).

Große Plasmabögen wurden prozedural modelliert: Kugeln wurden als Mograph-Klone über die Sonnenoberfläche verteilt und mit einem Shader-Effekt passend



© ZDF 2015

www.renderbaron.de

angeordnet. Mit dem Plug-in pCONNECTOR (tcastudios.com) wurden diese Klone dann verbunden, das resultierende Spline-Objekt wurde in ein Sweep-Objekt geworfen. Die so erzeugten röhrenförmigen Verbindungen zwischen den Klonen wurden dann mit einem Shader-basierten Displace-Deformer zu Bögen ausgebeult (Bild 11).

Eintauchen in die Sonne

In Min 00:11 der Animation tritt die bewegte Kamera in das Volumen der Sonne ein. Dieser Übergang erfolgt im Compositing mit einer Überblendung zu einer Sequenz, die vom technischen Ansatz her dem stellaren Nebel des letzten Artikels ähnelt. Wichtig ist dabei, dass für diesen Übergang die Endgeschwindigkeit der Außenkamera die gleiche Richtung und Lineargeschwindigkeit aufweisen muss wie der Start der Innenkamera.

Schnelles Rendering mit Standard-Renderern

Sample-basiertes volumetrisches Licht ist seit einer kleinen Ewigkeit Bestandteil von C4D, nämlich seit Release 5 (1998). Diese betagte Art von Samples wird von C4D natürlich anders gehandhabt als die modernen Samples von Flächenschatten, gestreuten Reflexionen, Ambient Occlusion und Co. Folglich ist das Rendern dieser Szene im älteren Standard-Renderer weitaus schneller als im jüngeren physikalischen Renderer.

Darüber hinaus wird der Standard-Renderer seit Release 19 (2017) durch die Intel-Embree-Technologie beschleunigt. Embree ist eine Raytracing-Bibliothek, die für die neuesten Intel-Prozessoren optimiert ist und das Rendering deutlich beschleunigt – in einigen Fällen bis zu 100% im Vergleich zu Release 18. Embree haucht dem guten alten Standard-Renderer quasi neues Leben ein.

Aber das ist noch nicht alles: Noises und prozedurale Shader haben standardmäßig eine hochqualitative SAT-Texturinterpolation, sodass für das Rendering mit dem Standard-Renderer keine Notwendigkeit besteht, ein Antialiasing zu verwenden. Um es auf den Punkt zu bringen: Der Standard-Renderer rendert diese Szene mit Lichtgeschwindigkeit – insbesondere im Hinblick auf die hier entstehende volumetrische Komplexität.

Fluffige Wolken mit Mograph

Die für stellare Nebel und Sonneneruptionen geschilderte Herangehensweise basiert auf dem Volumen einer sichtbaren volumetrischen Lichtquelle als „Aquarium“ für 3D-Noises. Während die Verwendung eines

solchen Licht-Containers recht einfach und unkompliziert ist, hat sie doch einen entscheidenden Nachteil: Alle geschaffenen Strukturen und Effekte sind tatsächlich sichtbares Licht ohne echte Opazität und ohne die Möglichkeit eines Schattenwurfs. Eine so erstellte Wolke wird also nie einen Schatten auf den Boden oder auf sich selbst werfen.

Um also eine schattenspendende fluffige Wolke für einen sonnigen Sommerhimmel zu erschaffen, müssen wir in eine Technik eintauchen, die zwar auch auf der Verwendung volumetrischer Shader basiert, jedoch einen anderen Container verwendet: einen Mograph Cloner.

Mograph – prozedurales Klonen und Animation

Mograph, das proprietäre Toolset von C4D, bietet Motion Designern eine umfangreiche Palette leistungsstarker Tools für prozedurale non-destruktive Animationen. Eine Kernfunktion von Mograph ist das Klonen von Objekten nach bestimmten Regeln, z.B. entlang eines Splines, in bestimmten Anordnungen oder auf der Oberfläche eines Objekts. Effektoren wie Zufall, Shader oder Zeit u.a. bringen Variation und Bewegung in das Klon-System und



Bild 15: Ein Farbverlauf im Leuchten-Kanal erzeugt eine einfache Textur für das Himmel-Objekt.

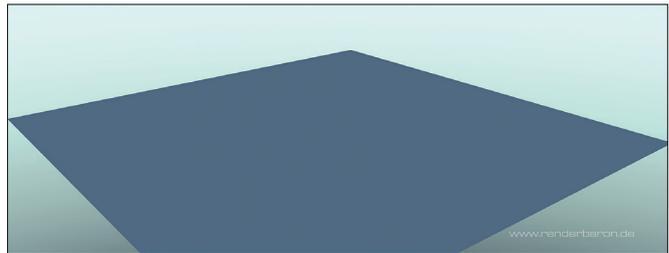


Bild 16: Ausgangspunkt zum Wolken-Bauen: unsere Szene mit Himmel-Objekt und Bodenplatte

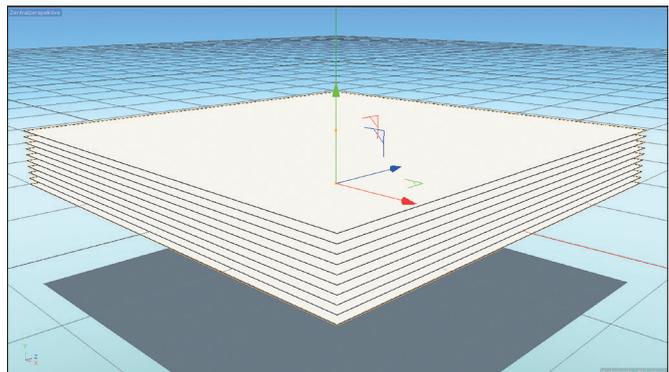


Bild 17: Grundprinzip der Wolken: Platten als Mograph-Klone dienen quasi als Volumen-Samples.

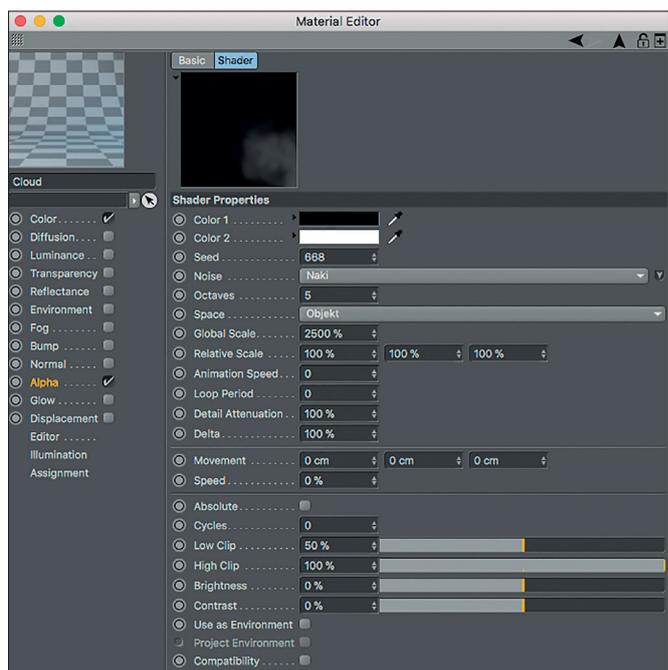


Bild 18: Der Wolken-Shader – ein einfaches Naki-Noise, zunächst im Raum Objekt

mit den Fields von Cinema4D R20 sind der Fantasie und der Komplexität quasi keine Grenzen mehr gesetzt.

Während Mograph eine riesige Palette von spannenden Funktionen bietet und so zum Spielen und Experimentieren einlädt, werden wir uns auf die wahrscheinlich langweiligste Funktion konzentrieren: das Klonen einfacher Ebenen (also Polygon-Platten) in einen linearen Stapel. Dies wird nämlich der Container für unsere 3D-Noises sein: ein Stapel geklonter Ebenen mit nur einem Polygon. Bevor wir in die darauf basierende Technik zur Erstellung von Wolken einsteigen, lasst uns sehen, was man daraus machen kann.

Fallstudie: ZDF „Terra X“, Schottland

Dieses Projekt für die TV-Dokureihe ZDF „Terra X“, Episode „Schottland – der Mythos der Highlands“ (Bild 13, 14, 15) zeigt die geografische Herkunft Schottlands und Englands. Volumetrische Wolken in den Close-up-Shots werden mit der Mograph-basierten Technik erzeugt, die wir uns nun in Form eines kleinen Tutorials ansehen werden.

Tutorial – fluffige Scheibchen-Wolken

Um diese folgend geschilderte Technik nicht nur zu verstehen, sondern auch in die Fingerspritzen zu bekommen, folge einem kurzen Tutorial.

Erstellen einer Umgebung

Erstelle eine neue Szene, erzeuge unter Hauptmenü > Erzeugen > Umgebung ein einfaches Himmel-Objekt. Erzeuge weiterhin ein neues Material, deaktiviere alle Materialkanäle, aktiviere den Leuchten-Kanal und erstelle darin einen schönen 2D-V-Farbverlauf in typischen Himmelsfarben, vielleicht wie in **Bild 15** zu sehen.

Platziere ein Ebene-Objekt mit 1x1 Segmenten und Abmessungen von 1.200x1.200cm bei -100 cm Y-Höhe der Welt und wende ein neues Material mit bläulicher Farbe auf die Ebene an. Diese Platte dient uns als Bodenebene. Erstelle eine Kamera und betrachte die Szene von einer flachen Draufsicht aus. Die leere Szene sollte ungefähr so aussehen wie in **Bild 16**.

Erstelle als Sonne ein unendliches Licht und wende vorerst Raytraced-Schatten (Hart) an (nur um einen schnellen Schatteneffekt zu erzielen, egal wie unrealistisch er ist).

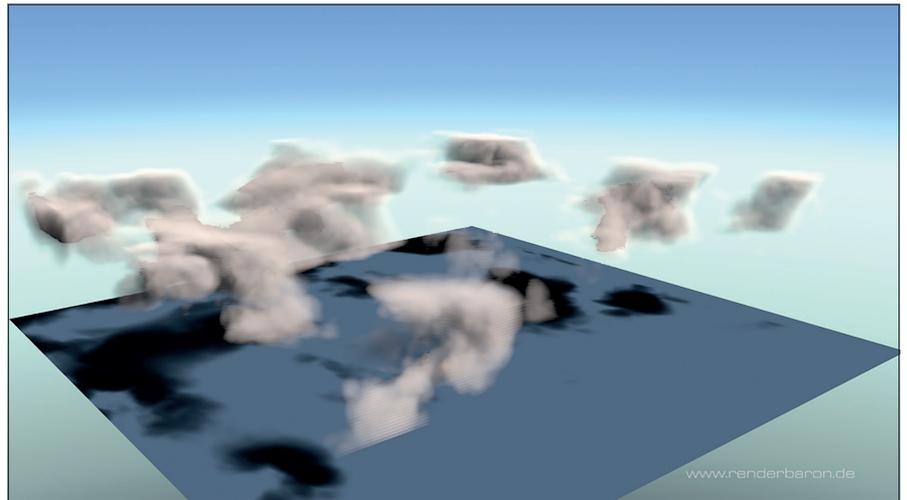


Bild 19: Das Naki im Raum Weit durchdringt den Mograph-Stapel. Es wird wolkelig – allerdings noch mit Artefakten.

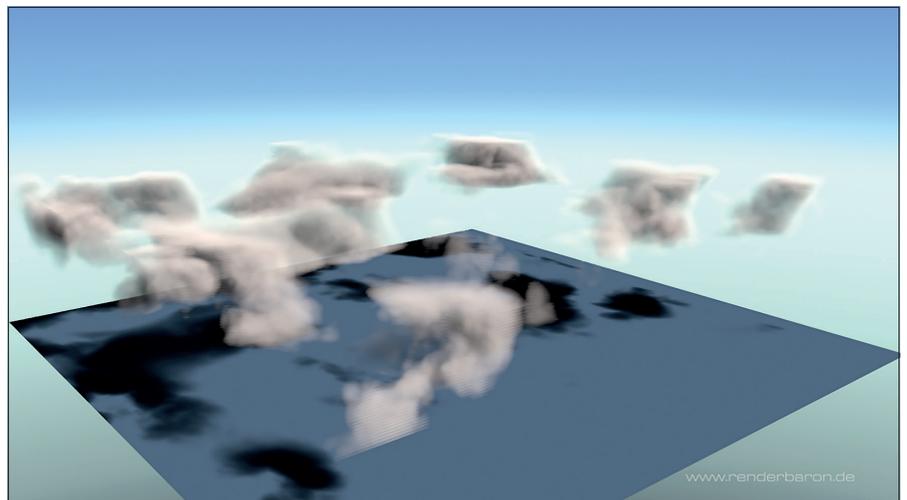


Bild 20: Eine Erhöhung der Strahltiefe in den Optionen der Rendervoreinstellungen entfernt Artefakte.

Gestapelte Ebenen

Erstelle einen Mograph-Kloner unter Hauptmenü > Mograph > Kloner. Ordne eine Kopie der Boden-Ebene dem Kloner unter. Sofort wird die Ebene dreimal entlang der Y-Achse des Kloners vervielfältigt. Klicke im Objektmanager auf den Kloner und sieh im Attribute-Manager seine Einstellungen an: Du wirst feststellen, dass der Modus des Klonens standardmäßig auf linear gestellt ist, genauso wie wir es wünschen. Stelle die Anzahl der Klone zunächst auf 10 und stelle den Entfernenmodus von Schritt auf Endpunkt. Den Abstand unter P.Y. stellst du auf 50 cm. Dadurch werden alle geklonten Ebenen entlang der Y-Achse der Welt innerhalb dieses Bereichs verteilt (**Bild 17**).

Erstellen eines Wolken-Materials

Lege ein neues Material an. Aktiviere im Material den Alpha-Kanal und erstelle darin einen Noise Shader. Wähle im Noise Shader

die Noise-Art Naki, wähle als Raum Objekt und justieren die globale Größe auf stramme 2.500%. Stelle das Clipping unten auf 50% und Clipping oben auf 100%. Dies verengt den Tonwertverlauf des Noises, sodass du schön voneinander unterscheidbare weiße und schwarze Bereiche erhältst (**Bild 18**). Dies sorgt im Alpha-Kanal für definierte transparente Bereiche (schwarz) und opake Bereiche (weiß).

Optimieren des C4D-Viewports

Bevor wir in der Wolkenerstellung fortfahren, optimieren wir kurz unseren Viewport, um erste Fassungen unserer volumetrischen Wolken in Echtzeit (!) im Viewport betrachten zu können. Wähle dazu im Viewport-Menü unter Optionen den Punkt „Enhanced OpenGL“ (falls nicht bereits aktiviert). Aktiviere dann Noise und Transparenz. Das Ergebnis sehen wir uns gleich an.

Jetzt wird's fluffig

Wende das vorhin erstellte Material auf deinen Kloner an. Du erhältst daraufhin 10 Ebenen mit genau der gleichen Noise-Textur – obwohl wir das Naki Noise doch im Raum Objekt angewendet haben! Das liegt daran,

dass der Kloner kein „echtes“ Objekt ist – das Noise im Raum Objekt referenziert stattdessen das Achsensystem jeder Ebene und wiederholt das Naki Noise mit jedem Klon. Um das zu ändern, gehen wir zurück zum Naki Noise und wählen Raum Welt. Jetzt erhalten wir eine Darstellung des Naki Noises, die von Ebene zu Ebene wechselt – so wie wir es wollen.

homogener wird das Ergebnis. Der Parameter „Anzahl“ im Kloner fungiert nun also als eine Art Sample-Anzahl.

Mit den vorhin optimierten Einstellungen des C4D-Viewports kannst du diese erste Fassung deiner volumetrischen Wolken in Echtzeit begutachten (!).

Erhöhen der Strahlentiefe

Erhöhe nun Schritt für Schritt deine Klon-Anzahl auf 50 und sieh, was passiert ... Bingo! Die Scheiben des Mograph Cloners – die Ebenen – dienen als eine Art räumlicher Samples für das 3D-Noise und erzeugen so volumetrische Wolken. Je mehr Klone du anwendest, desto

Beim Rendern erhältst du nun einige hübsche wolkenähnliche Strukturen, welche aber hier und da schwarze, scharfe Artefakte aufweisen (Bild 19). Dies liegt daran, dass die Strahlentiefe des Raytracers aufgebraucht ist. Das bedeutet, dass der Strahl des Raytracers nach dem Durchlaufen einer begrenzten Anzahl von Transparenzen (unsere geklonten Ebenen mit Alpha-Kanal) beendet wird und einen schwarzen Pixel rendert.

Um dies zu vermeiden, wähle unter Rendervoreinstellungen (Strg+B) den Eintrag „Optionen“ und passe den Parameter „Strahlentiefe“ an. Als Wert solltest du die Anzahl deiner Klone plus eins wählen. Das Ergebnis sieht schon besser aus (Bild 20). Da das häufigere Durchdringen von Transparenzen durch den Raytracer-Strahl die Renderzeit verlängert, verwenden wir vorerst einen Raytraced-Schatten für die in Schritt 01 erstellte Sonnen-Lichtquelle. Für das finale Rendern kannst du dann immer noch realistische Flächenschatten verwenden.

Vertikale Maskierung

Was wir deutlich sehen können, ist, dass alle Wolken oben und unten durch die vertikalen Grenzen unseres Cloners abrupt abgeschnitten werden. Um dies zu umgehen, maskieren wir unser 3D-Noise mit einem vertikalen 3D-Farbverlauf.

Klicke im Alpha-Kanal deines Wolkenmaterials neben Textur auf den Dreiecksknopf

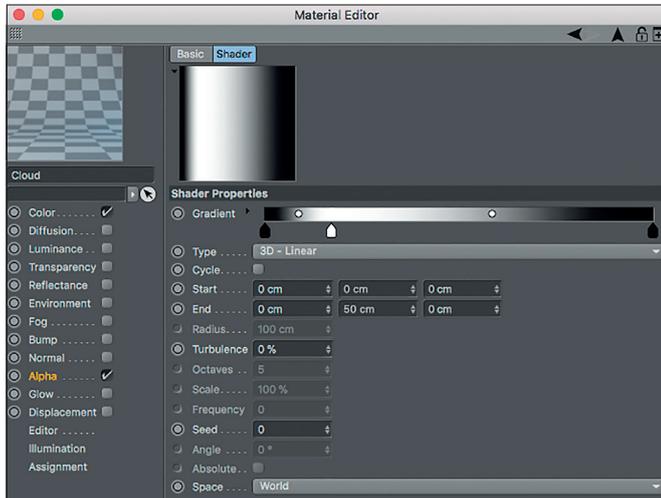


Bild 21: Ein linearer 3D-Verlauf in der Höhe des Mograph-Stapels sorgt für eine weiche vertikale Begrenzung.

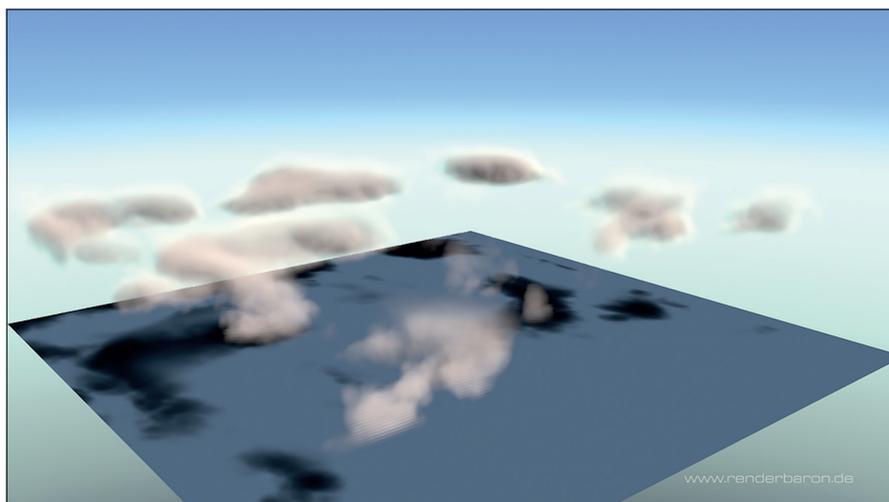


Bild 22: Die vertikale Begrenzung der Wolken an ihrer Unter- und Oberseite ist noch zu weich.



Bild 23: Eine kleinere, invertierte Version des Cloud-Noises liegt als Mask-Noise im Ebene-Shader oben auf.

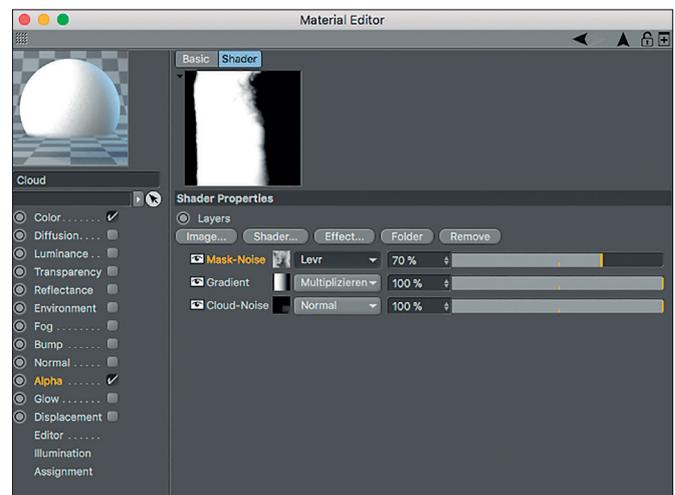


Bild 24: Das Mask-Noise im Ebenen-Modus Levr gibt dem Verlauf darunter mehr Struktur.

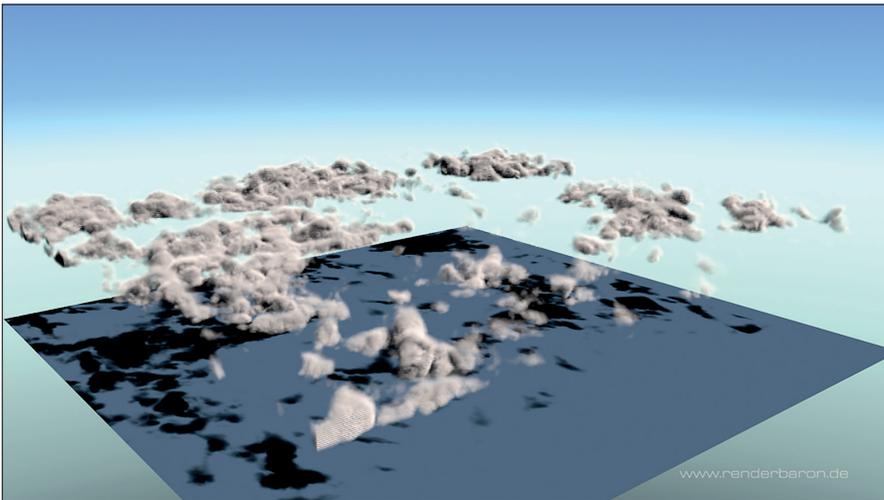


Bild 25: Die vertikale Begrenzung der Wolken hat nun eine schön definierte Struktur.

und wähle „Ebene“. Du hast nun deinen Noise-Shader in einen Ebene-Shader verschoben.

Gehe kurz zurück zu deinem Alpha-Kanal und deaktiviere die Checkbox „Alphabild“, um sicherzustellen, dass die Graustufeninformationen unseres Ebene-Shaders auch als Alpha-Informationen interpretiert werden. Doppelklicke im Ebene-Shader auf das Noise und nenne es Cloud-Noise. Klicke dann auf den Button „Shader“, erstelle einen Farbverlauf-Shader und ziehe ihn oberhalb des Cloud-Noises. Als Typ des Verlaufs wähle 3D-Linear.

Wir müssen uns nun überlegen, wie wir den Farbverlauf korrekt justieren, um unser Cloud-Noise von unten nach oben räumlich zu maskieren. Da die unterste Ebene unseres Kloners bei $Y=0$ liegt und der Kloner mit allen Ebenen eine Dicke/Höhe von 50 cm hat, stellen wir den 3D-Farbverlauf so ein, dass er entlang der Y-Achse bei Startpunkt 0 cm beginnt und bei Endpunkt 50 cm endet. Als Bezugssystem wählen wir Raum Welt. Der Verlauf geht so entlang der Welt-

Y-Achse von Schwarz bei 0 cm zu Weiß wieder zu Schwarz bei 50 cm (**Bild 21**). Zurück im Ebene-Shader stellen wir den Ebenen-Modus auf Multiplizieren. Jetzt können wir rendern.

Das Ergebnis ist nun viel besser, da wir das Cloud Noise nun räumlich von unten nach oben maskieren und so Kontrolle über seine vertikale Ausgestaltung haben (**Bild 22**). Allerdings ist die Maske viel zu weich und zu homogen. Anstatt nun unserem Farbverlauf eine Turbulenz zu verleihen, werden wir etwas Eleganteres damit anstellen.

Kopiere das Cloud-Noise und füge es als Kopie in deinen Ebene-Shader ein (rechte Maustaste, Copy Shader/Paste Shader). Doppelklicke auf das eingefügte Noise und nenne es Mask-Noise. Ziehe dann das Noise oberhalb des Farbverlaufs.

Ändere die globale Skalierung des Mask-Noises auf 500% und invertiere das Noise, indem du in ihm das Clipping unten auf 85% und das Clipping oben auf 0% einstellst. Gehe zurück in den Ebene-Shader und stelle den Ebenenmodus des Verlaufs

darunter auf Normal. Das Setup sollte nun so aussehen wie in **Bild 23**.

Stelle nun den Ebenenmodus des Mask-Noises auf Levr und sieh nach, was grob passiert: Eine Art Hochkontrast-Version des Mask-Noises wird vom Verlauf weggeschnitten. Verringere die Deckkraft des Mask-Noises auf 70%, um eine weichere Kante zu erhalten. Stelle den

Ebenen-Modus des Verlaufs wieder auf Multiplizieren (**Bild 24**). Vor dem Rendern justieren wir im Cloud-Noise den Parameter Clipping unten auf 20%. Dann rendern wir wieder. Das Ergebnis sollte in etwa aussehen wie in **Bild 25**.

Shadow Luminance

Die Wolken werden nun zunehmend realistischer, sehen aber noch grau und dunkel aus. Grund: Jeder Mograph-Klon wirft einen Schatten auf den darunterliegenden Klon. Um nun ausschließlich Schattenbereiche aufzuhellen, verwenden wir ein von mir erstelltes Shader Setup, um eine Art Selbstleuchten auf Schattenseiten von Objekten zu simulieren: Shadow Luminance.

Obwohl ich dieses Shader Setup ursprünglich entwickelt habe, um sehr diffuses Licht auf Schattenseiten von Objekten zu simulieren, kann es auch dazu verwendet werden, um eine Art Subsurface Scattering auf unseren Wolken zu simulieren.

Shadow Luminance besteht dabei aus drei wichtigen Komponenten:

- ▷ Die erste Komponente ist ein Ebene-Shader, der als Container dient.
- ▷ Die zweite Komponente ist ein Lumas Shader, der erkennt, wo Licht und wo Schatten ist. Lumas tut dies durch seinen Tab Shader, der sich wie der Farbkanal verhält und sagt: „Zeig mir, wo ich im Licht bin.“
- ▷ Die dritte Komponente ist ein Colorizer Shader, in den der Lumas reingeworfen wird. Im Colorizer wird dann der Farbverlauf auf Weiß-Schwarz gestellt, was den enthaltenen Lumas umkehrt zu „Zeige mich, wo ich im Schatten bin“.

Der so mit dem Lumas gefütterte Colorizer wird dann verwendet, um eine beliebige Farbe auf die Schattenseite eines Objekts zu maskieren und so eine leichte Aufhellung der Schattenbereiche zu erzeugen. Setzen wir das in die Tat um.

Aktiviere den Leuchten-Kanal deines Wolkenmaterials. Erstelle einen Ebene-Shader. Erstelle innerhalb des Ebene-Shaders einen Lumas Shader. Deaktiviere alle Glanzaspekte des Lumas Shaders. Wähle unter Tab Shader eine Beleuchtung von 100% und stelle die Farbe auf ein helles Weiß ein. Geh zurück in den Ebene-Shader und lege den Lumas in einen Colorizer Shader, indem du mit der rechten Maustaste auf den Lumas klickst und Colorizer wählst. Im Inneren des Colorizer Shaders stellst du seinen Farbverlauf auf Weiß-Schwarz ein.

Gehe zurück in den Ebene-Shader, klicke auf den Button „Shader“ und erzeugen einen Farbe-Shader. Zieh ihn

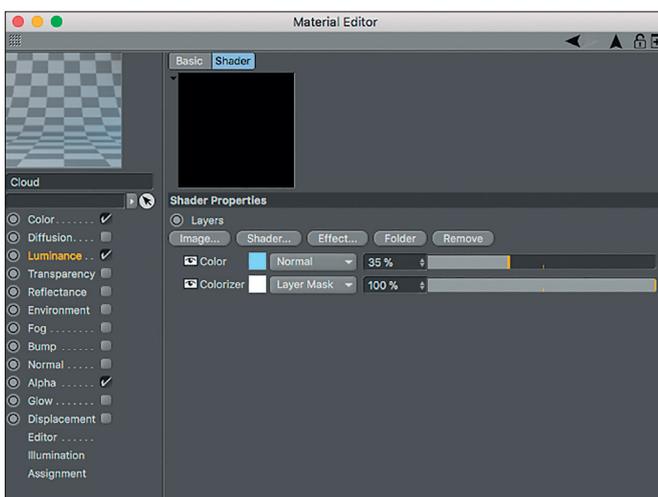


Bild 26: Aufhellung der Schattenseiten der Wolken durch Shadow Luminance



Bild 27: Gesteigerte Lichtempfindlichkeit der Wolken durch 150% diffuse Stärke

oberhalb des Colorizers. Stelle die Farbe des Farbe-Shaders auf ein helles Blau. Zurück im Ebene-Shader stellst du die Deckkraft der Farbe-Ebene auf 8% ein. Wähle für den darunterliegenden Colorizer den Ebenenmodus Ebenenmaske.

Das Setup sollte nun so aussehen wie in **Bild 26**. Voilà! – Du maskierst jetzt ein helles Blau ausschließlich auf die Schattenseiten deiner Wolken.

Erhöhen der Lichtempfindlichkeit

Wenn du dir den Farbkanal unseres Wolkenmaterials ansiehst, wirst du sehen, dass das Dropdown-Menü auf Lambert eingestellt ist. Lambert ist eine sogenannte BSDF, eine Bidirectional Scattering Distribution Function – oder einfach ausgedrückt: eine Funktion, die beschreibt, wie Licht über die Oberfläche eines Objekts verteilt wird, von seinem hellsten bis zum sogenannten Terminator, der Tag-Nacht-Grenze. Lambert

simuliert eine perfekt diffuse Oberfläche, während die andere verfügbare BSDF Oren-Nayar zusätzlich Mikrofacetten für einen satinierten, aufgerauten Look berechnet. Während Lambert eine gute Wahl für unsere hellen Wolken ist, hat Oren-Nayar einen wichtigen Vorteil: Die Parameter diffuse Stärke und Rauigkeit. Mit Rauigkeit kannst du nahtlos zwischen Lambert- (0% Rauigkeit) und Oren-Nayar-Verhalten (100% Rauigkeit) mischen. Und mit diffuser Stärke kannst du quasi die Lichtempfindlichkeit einstellen. Mit einer Kombination von 0% Rauigkeit und 200% diffuser Stärke haben wir also eine Lambert-BSDF mit erhöhter Lichtempfindlichkeit und dadurch erhöhtem Albedo (**Bild 27**).

Mit einer weiter erhöhten Klonanzahl (80), angepasster Strahlentiefe, Shadow Luminance für das Bodenmaterial und Flächenschatten für die Sonnen-Lichtquelle sollte dein Endergebnis wie in **Bild 28** aussehen.

Fazit

Fluffige Scheibchen-Wolken sind eine Frage der Klonzahl und der Strahlentiefe. Je homogener das gewünschte Ergebnis, desto länger dauert das Rendern. Insbesondere bei der Anwendung realistischer Flächenschatten werden Renderzeiten wieder er-

Maxon Authorized Training Center (ATC)

Studio renderbaron ist Maxon Authorized Training Center (ATC) und bietet Schulungen zu C4D R20 an – mit herausragendem Standard bei Trainingsqualität, Ausstattung und Service.

Mit dem Maxon Quickstart Training „Shading, Lighting & Rendering“ haben erfahrenere C4D-Anwender die Möglichkeit, sich mit diesem umfassenden Themengebiet tiefgehend auseinanderzusetzen.

Das Training widmet sich dem professionellen Umgang mit Oberflächeneigenschaften, Licht und Bildberechnung in C4D. Schwerpunkte des Trainings sind Analyse und Umsetzung natürlicher Lichtsituationen mit manuellen Methoden sowie das Erstellen überzeugender Shading-Lösungen. Außerdem werden Techniken zur effizienten Bildberechnung, automatisierte Beleuchtungsmethoden (GI, HDRI) und jede Menge Praxis-Tipps vermittelt.

Das 3-tägige Training stammt aus der Feder von renderbaron-Inhaber und Maxon Lead Instructor Marc Potocnik und bietet eine geballte Ladung Wissen und Expertise aus über 22 Jahren Branchenerfahrung. Es umfasst ein Skript von ca. 190 Seiten, zahlreiche Beispieldateien sowie eine Handvoll ausgewachsener Praxisprojekte. Im deutschsprachigen Raum kann das Training in Form einer Individual- oder Firmenschulung ausschließlich bei renderbaron gebucht werden. Anfragen können gerne an marc@renderbaron.de gerichtet werden.

höht. Daher besteht der goldene Mittelweg in einem Gleichgewicht aus niedriger Klonzahl/Strahlentiefe und homogenen Ergebnissen – gehe mit beidem behutsam um!

Der physikalische Renderer ist in diesem Fall die beste Wahl, da er eine qualitativ hochwertige Antialiasing-Funktion, schnellere Berechnung von Flächenschatten-Samples und ebenfalls Intel-Embree-Beschleunigung bietet.

Übrigens: Eine Live-Fassung dieser kleinen Artikelreihe findest du als Mitschnitt vom Maxon Supermeet 2018 unter www.renderbaron.de/publikationen. > ei

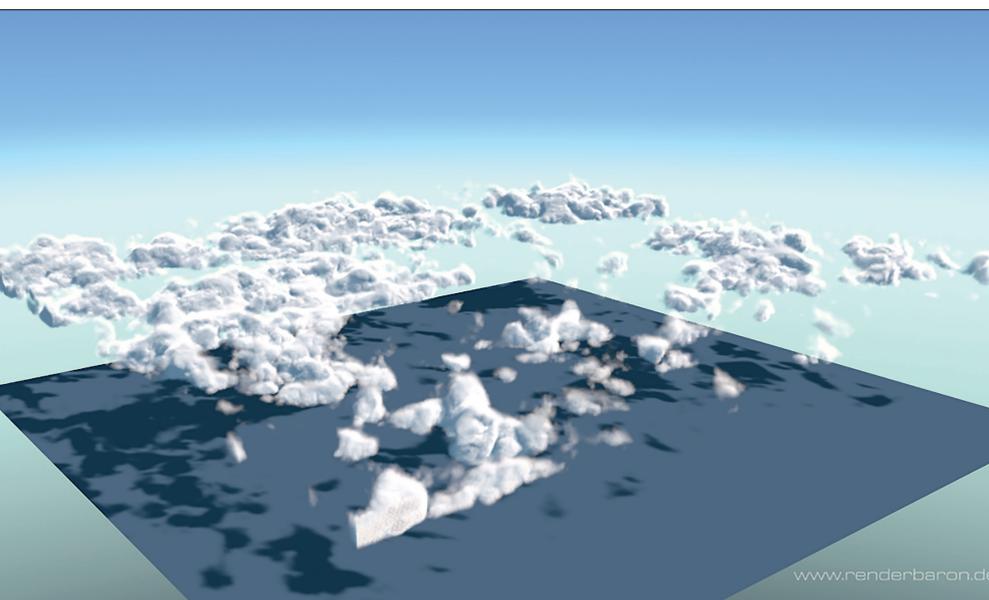


Bild 28: Das Endergebnis: dichte, fluffige Schäfchen-Wolken



Marc Potocnik ist Diplom-Designer (FH) und Inhaber des Animationsstudios renderbaron in Düsseldorf. renderbaron realisiert seit 2001 hochwertige 3D-Animationen für renommierte Kunden wie ZDF, Audi, BMW und bietet als Maxon Authorized Training Center (ATC) Trainings zu C4D R20 an. Marc Potocnik ist Maxon Lead Instructor und Autor des Maxon Quickstart Trainings „Shading, Lighting & Rendering“. Er teilt sein Wissen außerdem in Form von Fachvorträgen auf internationalen Branchen-Events wie der Siggraph, FMX, IBC etc. Außerdem ist Marc Alpha- und Beta-Tester für C4D.