ISSN 1433-2620 > B 43362 >> 23. Jahrgang >>> www.digitalproduction.com Publiziert von DETAIL Business Information GmbH

2019

PRODUCTION **2** 

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

DIGITAL PRODUCT

MAGAZIN FÜR DIGITALE MEDIENPRODUKTION

...Z | APRIL 02:2019

**Plug-ins** Oder soll es länger dauern als unbedingt nötig?

**animago!** Monster mit den Gewinnern La Noira & Mackevision und vieles mehr! BMD Pocket 4K, Caracter Animator, Flame, Arri ...



Stellarer Nebel aus ZDF "Terra X", Episode "Eine Frage der Zeit" (https://vimeo.com/renderbaron/ txeinefragederzeit, Min 00:30): Ein einziges sichtbares volumetrisches Licht dient hier als Aquarium für 3D-Noises und weitere volumetrische Shader, die als Textur auf der Lichtquelle liegen – Komplexität leicht gemacht.

www.renderbaron.de

# Wolkig bis feurig – Volumetrische Effekte in Cinema 4D

# Teil 1: Die Magie der sichtbaren Lichter

Nebel, Plasma etc. sind eine häufige Aufgabe in der täglichen Arbeit eines 3D-Artists. Technische Ansätze für die Erstellung in Cinema 4D wären üblicherweise Simulationen mit Plug-ins wie z.B. Turbulence FD und X-Particles oder aber der Einsatz des alternden, auf Voxel basierenden Pyro-Cluster-Systems von Cinema 4D.

Im Folgenden werden wir nun jedoch Arbeitsweisen kennenlernen die auf sehr einfache und dennoch elegante Weise volumetrische Effekte nur mit den Onboard-Tools von Cinema 4D ermöglichen – und das ganz ohne die eingangs genannten üblichen Techniken. Dabei widmen wir uns zweierlei Einsatzmöglichkeiten mit steigender Komplexität: 1. dünnen Wolken und Nebelfeldern sowie 2. stellaren Nebeln.

## Dünne Wolken und Nebelfelder

Der einfachste und direkteste Ansatz zur Erzeugung grundlegender atmosphärischer Effekte ist die Verwendung von Cinema 4Ds In dieser zweiteiligen Artikelreihe eröffnen wir dem gewandten Cinema-4D-Nutzer tiefgehende Einblicke in die Erstellung volumetrischer Effekte mit den Onboard-Tools von Cinema 4D – und das ganz ohne Plug-ins, Partikel oder Pyro Cluster. Im erste Teil der Reihe widmen wir uns sichtbaren Lichtern und deren Texturierung mit volumetrischen Shadern. von Marc Potonic

eingebauten Lichtquellen als eigentlichen Wolken oder Nebelfeldern. Mit einer Lichtquellenoption zur Erzeugung von nebligem sichtbarem Licht und einigen eingebauten Noise-Funktionen können Sie im Handumdrehen dünne, nicht schattenspendende Wolken oder Nebelfelder erzeugen. Aber bevor wir ins Detail gehen, sollten wir uns einige der grundlegenden Funktionen von Lichtquellen in Cinema 4D ansehen. Sobald Sie in Cinema 4D eine Lichtquelle erzeugen, zeigt der entsprechende Attribute-Manager eine Reihe von Reitern (Tabs) an (Bild 01):



Bild 01: Ein Überblick über die grundlegenden Parameter einer Lichtquelle: Lichtquellenart, Schattenart und Art des sichtbaren Lichts. Zusätzlich eingeblendet sind die Tabs für Sichtbarkeit (Visibility) und Noise.

ART

Bild 02 und 03: Ein paralleler Spot mit sichtbarem volumetrischem Licht: ein zu hoher Wert im Parameter Sample-Dichte (Tab Sichtbarkeit) führt zu Vergröberungs-Artefakten (links), ein geringer Wert hingegen erzeugt mehr Samples und homogenere Ergebnise (rechts).



Bild O4: Sichtbare volumetrische Lichter sollten in kompletten Szenen immer als separates Take "auf Schwarz" gerendert werden. Die Funktion "Einziges Material" ist hier maßgeblich. Das Ergebnis wird dann in der Postproduktion auf das separat erstellte "vollständige" Rendering mit einem Modus wie z.B. "Negativ Multiplizieren" aufgetragen.

#### Lichtquelle – Tab Allgemein

Im **Tab Allgemein** sind die Farbauswahl und die Intensitätsskala die zunächst auffallenden Parameter. Der Typ der Lichtquelle kann im Dropdown-Menü **Typ** weiter unten ausgewählt werden. Im Folgenden werden die Arten von Lichtquellen nach ihrem Grad der Verwandtschaft zueinander aufgelistet. Die Form der Lichtemission bestimmt dabei die Form ihres sichtbaren/volumetrischen Effekts:

- Omni: konzentrische Lichtemission
- Spot: konische Lichtemission
- Quadratischer Spot: pyramidenförmige Lichtemission
- Paralleler Spot: parallele, zylindrische Lichtemission
- Quadratischer paralleler Spot: parallele, quaderförmige Lichtemission
  Parallel: parallele Lichtemission
- Denotion: parallele Lichtemission von
- einem unendlich weit entfernten Punkt aus
- Fläche: räumlich ausgedehnte Lichtemission mit einer bestimmbaren geometrischen Form
- IES: simuliert das herstellerspezifische Verhalten von Beleuchtungssystemen mit photometrischer Intensität

Das Dropdown-Menü **Schatten** bietet eine Auswahl aus drei Arten der Schattenerzeugung:

- Shadowmaps (weich): erzeugt eine Schattentextur aus Sicht der Lichtquelle mit wählbarer Auflösung, Weichzeichnungsradius und Nähe-Toleranz (Bias).
- Raytraced (hart): erzeugt einen unendlich harten Schatten aus der Sicht der Lichtquelle. Dies ist die bei Weitem älteste und unrealistischste Art der Schattenerzeugung.
- Fläche: die einzige Art von Schatten, die je nach Entfernung zum schattenwerfenden Objekt schärfer oder diffuser wird. Diese Schattenart ist hauptsächlich für die Verwendung mit Flächenlichtern oder unendlichem Licht gedacht.

Um dünne Wolken oder Nebelfelder zu erzeugen, müssen wir keine Schatten verwenden, da die sichtbare Struktur von der Lichtquelle selbst kommt. Für die Erstellung fluffiger, schattenwerfender Wolken werden wir im nächsten Artikel jedoch auf Flächenschatten zurückkommen. Das Dropdown-Menü Sichtbares Licht wiederum erzeugt den eigentlichen Effekt, mit dem wir dünne Wolken oder Nebel simulieren: neblige und dadurch sichtbare Luft. Es bietet drei Modi:

- Sichtbar: Objekte im sichtbaren Licht werfen keinen volumetrischen Schatten, auch wenn eine Art Schatten aktiviert ist.
- Volumetrisch: Objekte im sichtbaren Licht werfen volumetrische Schatten – auch wenn kein Schatten aktiviert ist (!).
- Invers volumetrisch: Sichtbares Licht wird nur im volumetrischen Schatten von Objekten erzeugt.

# Lichtquelle – Tab Sichtbarkeit

Das **Tab Sichtbarkeit** referenziert auf das Dropdown-Menü **Sichtbares Licht** im **Tab Allgemein** und bietet zahlreiche Parameter zur Definition des sichtbaren Lichts und dessen Sichtbarkeit entlang der Achse oder des Radius der Lichtquellen:

- Axiale Abnahme verringert die Sichtbarkeit entlang der Z-Achse der Lichtquelle (z.B. bei Spots).
- Radiale Abnahme verringert die Sichtbarkeit entlang des Radius der Lichtquelle (z.B. bei parallelen Spots).
- Innere/Äußere Distanz definiert den Start- und Endpunkt der Stärke des sichtbaren Lichts ausgehend von seinem Ursprung. Diese Abnahmeparameter arbeiten unabhängig von den Abnahme-Parametern im Tab Details – die Anfasser dieser Parameter können im Editor jedoch leicht miteinander verwechselt werden.
- Helligkeit steuert die Intensität des sichtbaren Lichts.
- Staub fügt dem sichtbaren Licht bei niedrigen Helligkeitswerten eine schwarze Komponente hinzu, die den Eindruck einer deckenden Staubwolke erzeugt.
- Mit Relative Größe können Sie das sichtbare Licht unproportional und unabhängig von seiner tatsächlichen Form zusammendrücken und dehnen (z.B. bei Punktlichtern).

Wenn im **Tab Allgemein**/Dropdown-Menü Sichtbares Licht die Option Volumetrisch ausgewählt ist, erzeugt das sichtbare Licht volumetrische Schatten – auch wenn für die Lichtquelle kein Schatten aktiviert ist (!).





Bild 05 und 06: Die Notwasserung des Fluges US Airways 1549 im Hudson River für ZDF "Leschs Kosmos", Episode "Albtraum Fliegen" (https://vimeo.com/renderbaron/ lekoalbtraumfliegen] – simple sichtbare Lichter mit Noise-Funktion sorgen im Vordergrund über Governor Island für dünne Cumulus-Wolken. Dieselbe Technik wurde in der Sequenz zum Funktionsprinzip von Radar angewendet (im Bild rechts).



Bild 07: Definierte Wolkenstrukturen lassen sich im Tab Noise mittels starker negativer Helligkeit (-350 %) und starkem positiven Kontrast (700%) herausarbeiten.



Bild 08: Animation "Steigenkogel" (https://vimeo.com/renderbaron/steigenkogel) die Nebelfelder in der Distanz sind nichts anderes als sehr flache Punktlichter mit sichtbarem Licht, keiner Lichtabstrahlung und nur leichter Noise-Funktion.

#### Samplebasierte Berechnung

Die Auflösung des sichtbaren volumetrischen Lichts wird im Tab Sichtbarkeit durch den Parameter Sample-Dichte bestimmt. Bei diesem Parameter bzw. seiner Benennung muss man ein wenig um die Ecke denken: Die Sample-Dichte muss reduziert werden, um die Sample-Anzahl zu erhöhen. Ein hoher Wert (niedrige Sample-Dichte) beschleunigt das Rendern, kann aber zu scheibenartigen Artefakten im Licht-Volumen führen. Für ein homogenes Ergebnis benötigen Sie eine höhere Sample-Dichte, also einen niedrigeren Wert, z.B. 5 m statt der voreingestellten 25 cm. Dies kann das Rendering jedoch verlangsamen siehe Bild 02 (niedrige Sample-Dichte) und Bild 03 (hohe Sample-Dichte).

#### Tipp zum Rendern

Besonders wenn Sie volumetrisches Licht und ein komplexes Szenen-Setup (komplexe Objekte, hochauflösende Texturen, starkes Antialiasing) kombinieren, kann das Rendering ordentlich verlangsamt werden. Daher sollten Sie volumetrische Lichter immer separat auf Schwarz rendern und dann mit Ihrer Szene im Compositing kombinieren. So können Sie ordentlich Renderzeit sparen. Cinema 4Ds Take-System und die Option "Einziges Material" zum Überschreiben aller Materialien mit nur einem Material (Render-Voreinstellungen) bieten dazu die idealen Werkzeuge (Bild 04). Mehr Infos zu diesem Vorgehen finden Sie mit den genannten Stichwörtern in der Cinema-4D-Hilfe.

Mit den bis hierhin vorgestellten Parametern sind wir bereits in der Lage, Effekte wie Suchspots, Halos etc. zu erzeugen. Für Wolken und Nebel jedoch müssen wir unserem sichtbaren Licht vor allem eines verleihen: Unregelmäßigkeit. Zu diesem Zweck sehen wir uns nun das **Tab Noise** an

#### Tab Noise

Das Tab Noise verleiht Lichtquellen zufällige Unregelmäßigkeiten in Bezug auf Sichtbarkeit und/oder Beleuchtung. Ersteres variiert sichtbares Licht (falls aktiviert), Letzteres rein den Beleuchtungseffekt. Für beide Zwecke stehen 4 Noisefunktionen zur Verfügung: Noise, weiche Turbulenz, harte Turbulenz und wellenförmige Turbulenz.

Diese Noises können in Bezug auf Oktaven (Anzahl der Rechendurchgänge und damit Grad der Details), Helligkeit und Kontrast angepasst werden. Größe definiert die Größe des Noises nur im Bezug auf sichtbares Licht, Beleuchtungsgröße definiert die Größe des Noises nur in Bezug auf Beleuchtung.

Eine Bewegung des Noises kann über die Parameter **Geschwindigkeit** (Bewegung in sich), **Wind** (Richtung einer gleichmäßigen Bewegung) und **Wind-Geschwindigkeit** (Geschwindigkeit dieser gleichmäßigen Bewegung)

eingestellt werden.

Die Checkbox Lokale Koordinaten bestimmt, ob ein globales oder lokales Noise verwendet wird. Falls deaktiviert, werden bewegte Lichtquellen durch ein globales Noise bewegt, falls aktiviert (voreingestellt), nehmen bewegte Lichtquellen das Noise quasi mit.

Da sichtbare Lichtquellen mit Noise ein ziemlich altes, aber immer noch aktuelles Feature von Cinema 4D sind, stehen die vier Arten des Noise nicht in direktem Zusammenhang mit den Onboard-Noise-Shadern.

#### Fallstudie – Albtraum Fliegen

Dieses Projekt für die TV-Dokureihe ZDF "Leschs Kosmos" inszeniert in mehreren



Animationen die verborgenen Gefahren der Luftfahrt. Für die beiden Sequenzen über die Notwasserung des US-Airways-Fluges 1549 im Hudson River und über das Funktionsprinzip von Radar verwendeten wir einfache Punktlichter für dünne, turbulente Wolken im Vordergrund (Bild 05, Bild 06).

Die Punktlichter wurden in beiden Fällen auf Sichtbar, nicht aber auf Volumetrisch eingestellt. Das Kontrollkästchen Beleuchtung im Tab Allgemein wurde deaktiviert, um zu verhindern, dass die Wolken Licht ausstrahlen. Die Punktlichter wurden durch den Parameter Relative Größe im Tab Sichtbarkeit abgeflacht.

Im **Tab Noise** wurde eine harte Turbulenz als Noise verwendet. Der Kniff zum Erstellen definierter Wolkenstrukturen liegt dabei in der Kombination einer starken negativen (!) Helligkeit und einem starken positiven Kontrast (**Bild 07**).

Die dichten, fluffigen Wolken in den beiden Szenen wurden damals mit dem Plugin Ozone Cloudfactory 2015 erstellt – ein Plug-in für Cinema 4D R17, das zwar die Erstellung beeindruckender Wolkenfelder ermöglichte, jedoch sperrig zu handhaben war. Das Plug-in wurde vom Hersteller e-on Software Ende 2018 eingestellt. Ein detailliertes Making-of zum Projekt findet sich als Aufzeichnung meines Talks auf der Siggraph 2015 in Los Angeles hier: https://www.cineversity.com/vidplaylist/ r17\_siggraph\_2015\_rewind/.

#### Fallstudie – Steigenkogel

Dieses freie Projekt basiert auf der Intel®-Benchmark-Szene "Mountainvista", die ich im Frühjahr 2018 für Intel erstellt habe (https://vimeo.com/renderbaron/mountainvista). Die Landschaftsgestaltung der Szene ist dabei vollständig prozedural – von Felsen, Kieseln, Wurzeln und Schmutz bis hin zu Tälern, Bergen und Wolken.

Große Nebelfelder in der Ferne wurden mit der gleichen einfachen Technik wie oben beschrieben erstellt: große, flache Punktlichter mit nur sichtbarem Licht und einer aktivierten harten Turbulenz im Tab Noise (Bild 08).

#### Fazit

Nur mit Punktlichtern, sichtbarem Licht und einigen kontrastreichen Noise-Funktionen können dünne, nicht schattenspendende Wolken oder Nebelfelder erzeugt werden. Dieses Vorgehen ist dabei absolut einfach zu beherrschen und steht mit nur wenigen Klicks jederzeit zu Verfügung.

Die hauptsächliche Einschränkung dieser Technik ist die nur sehr grundlegende Verwendung von 4 Arten von Noise, die zudem weder eingefärbt noch mit praktischen Cinema-4D-Shadern kombiniert werden können. Sehen wir uns daher einen weiteren technischen Ansatz an, mit dem wir uns von diesen Einschränkungen lösen können.

## Stellare Nebel

SCIENCE & EDUCATION

Ein fortgeschrittenerer Ansatz zur Erstellung volumetrischer Effekte basiert auf dem Prinzip der Texturierung sichtbarer volumetrischer Lichter: Sie können jedes Material problemlos auf eine Lichtquelle anwenden – wenn der aktive Transparenzkanal z.B. einen 3D-Noise-Shader enthält, verwendet das Volumen der Lichtquelle diesen Shader zur dreidimensionalen volumetrischen Texturierung. Die Lichtquelle wird so quasi zu einem Aquarium für jeden volumetrischen Shader, der innerhalb des Transparenz-Kanals des Materials stattfindet! Wichtig: Dieses Prinzip funktioniert ausschließlich mit sichtbarem volumetrischem Licht, sichtbares Licht alleine lässt sich nicht volumetrisch texturieren.

Mit so einem Licht-Container ist es einfach, komplexere Strukturen wie Gasfelder oder stellare Nebel zu erstellen. Die Methode der Texturprojektion im Textur-Tag spielt dabei nur eine untergeordnete Rolle – solange Sie dreidimensionale Shader 3D-Farbverläufe oder Noise Shader im **Raum: Objekt** oder **Raum: Welt** verwenden.

Sehen wir uns zunächst ein einfaches Beispiel an: In **Bild 09** wird ein 2D-Tiles-Shader als Textur auf einem parallelen Spot mit sichtbarem volumetrischem Licht angebracht. Im Textur-Tag wurde die Projektion auf **Fläche-Mapping** gestellt. Die 2D-Textur folgt dabei dem **Fläche-Mapping** und durchdringt recht langweilig das Volumen des Spots.

In **Bild 10** ist alternativ ein 3D-Noise-Shader auf den Spot aufgebracht. Das **Fläche-Mapping** hat nun keine Wirkung mehr, da der Noise Shader durch seinen Parameter **Raum: Objekt** im Bezug zur Lichtquelle dreidimensional bzw. volumetrisch wirkt (Noise-Parameter, siehe unten).

Eine der Einschränkungen der eingangs geschilderten Technik für dünne Wolken war ja die strikte Verwendung von nur 4 einge-



Bild 11: ZDF "Terra X", Episode "Planet der Wälder" (https://vimeo.com/ renderbaron/txplanetderwaelder, Min 1:45): ein großflächiges Turbulenz-Noise sorgt für dünne Cumulus-Wolken im Vordergrund.



Bild 12: Animation "Steigenkogel" (https://vimeo.com/ renderbaron/steigenkogel) – ein groß skaliertes Luka-Noise erzeugt dünne Cirrus-Wolken in großer Höhe.

bauten Noise-Funktionen. Um Cinema 4Ds komplexere Noise Shader als 3D-Textur für volumetrische Lichter zu nutzen, ist es wichtig, sich zunächst mit den Grundlagen von Noise Shadern vertraut zu machen:

#### Noise Shader – systematischer Zufall

Naturphänomene wie Wolken, Gestein oder Wasseroberflächen zeichnen sich durch zufällige Strukturen mit unterschiedlichem Verhalten und unterschiedlicher Komplexität aus. In der Computergrafik nähern sich Noise Shader solchen natürlichen Strukturen. Mit Noise ist es möglich, Oberflächen eine zufällige, aber reproduzierbare Unregelmäßigkeit zu verleihen und sie so analoger und natürlicher erscheinen zu lassen. Mithilfe von Noise können alle Aspekte eines Materials moduliert werden. Im Vergleich zu 2D-Texturen bieten Noises den entscheidenden Vorteil, dass sie auch dreidimensional auf Objekte angewendet werden können, sodass alle Fragestellungen einer korrekten Texturprojektion oder UVW-Abbildung entfallen.

Der US-amerikanische Mathematiker Ken Perlin (https://mrl.nyu.edu/-perlin/) war der Erste, der sich der Entwicklung eines Noise Shaders widmete. 1981 war er als Mitarbeiter von MAGI in Elmsford, New York an der Entwicklung des Disney-Klassikers TRON beteiligt. Mit seinen bahnbrechenden Arbeiten zu Noise Shadern wollte Perlin den erzeugten Objekten ein weniger perfektes und dafür natürlicheres Aussehen verleihen. 1985 veröffentlichte Perlin ein Siggraph-Paper über das daraus resultierende Perlin Noise, 1997 erhielt er für seine Leistungen einen Oscar.

Durch verschiedene Weiterentwicklungen (z.B. Steven Worleys Cell-/Voronoi-Noise von 1996) fanden bereits 1998 eine Handvoll Noise Shader ihren Weg in Cinema 4D Release 5. Als Teil des von David Farmer entwickelten Shader-Plug-ins **Smells Like Almonds** (SLA) standen ab 2001 zwei Dutzend komplexere Noise Shader für Cinema 4D zur Verfügung, die auf exotische Namen wie Poxo, Luka, Sema oder Pezo hörten. Seit Cinema 4D Release 7.2 sind sie fester Bestandteil von Cinema 4D und in erweiterter Fassung bis heute an Bord.

Mit dem klassischen Kanal-basierten Materialsystem von Cinema 4D stehen Noise Shader als Channel Shader zur Verfügung und finden sich im Shader-Dropdown-Menü des Materialeditors unter Noise. Ab Release 20 von Cinema 4D sind Noise Shader auch als Nodes des neuen Node-basierten Materialsystems erhältlich.

Sie finden sie mit dem Stichwort Noise im Asset Manager des Node Editors. Da die in diesem Artikel gezeigten Kundenprojekte



mit Cinema 4D R19 erstellt wurden, werden wir uns auf die Channel-Shader-Version konzentrieren.

#### Perlin, Turbulence, Luka & Co.

Die Noise Shader von Cinema 4D haben sehr individuelle Eigenschaften und sind für verschiedene Zwecke prädestiniert. Werfen wir einen Blick auf ein paar Beispiele:

- Noise als Standardwahl ist nichts anderes als das Ur-Noise Perlin: Alle Komponenten sind von gleicher Größe, Verläufe zwischen hell und dunkel werden durch weiche Gradienten dargestellt. Höhere Oktaven, d.h. weitere Rechendurchgänge sind nicht verfügbar.
- Der nächste Verwandte Turbulence ist im Grunde ein Noise mit zusätzlichen Oktaven und kann dazu verwendet werden, um dünne, turbulente Wolken zu erzeugen (Bild 11).
- Luka mit seinen lebhaft unterschiedlichen, teilweise verquirlten Details bietet eine gute Grundlage für feine, dünne Wolken (Bild 12).

#### Eine ungleiche Verwandtschaft

Die Noise Shader scheinen auf den ersten Blick eine lose Ansammlung von Charakteren zu sein, aber offensichtlich sind sie enge Verwandte: **Turbulenz** mit nur einer Oktave sieht **Noise** zum Verwechseln ähnlich (testen Sie es mal!), **FBM** scheint ein höherkontrastiges Geschwister von **Turbulenz** zu sein, und die verschmierten Versionen wie **Wavy Turbulence, VL Noise** etc. tragen die Verwandtschaft schon im Namen (**Bild 13**). All diese Ähnlichkeiten haben einen Grund: Alle Noises in Cinema 4D sind die Nachkommen von einem Elternpaar: **Perlin** und **Voronoi**.

Für die Auswahl und Charakterisierung von Noises bietet Cinema 4Ds Hilfe um-

fassende und großformatige Übersichtsbilder. Eine zusätzliche Beschreibung der Shader finden Sie in der etwas älteren Noise Reference von cg.tutsplus.com unter bit.ly/2o8Jr6L.

#### Verwendung von Noise Shadern

Egal ob Sie Noises als Kanal-Shader oder als Nodes verwenden – Noises folgen einer prinzipiellen Funktion und haben fast alle Parameter gemeinsam: Die Art des Noises wird über das **Dropdown-Menü Noise** ausgewählt. Der Parameter **Oktaven** gibt die Anzahl der Berechnungsdurchgänge an und fügt so mit höherer Anzahl Details hinzu. Es folgen Parameter für **Größe**, **Animationsgeschwindigkeit** und Gradation (**Clipping**, **Helligkeit** etc.).

Noise wird in verschiedenen Referenzsystemen angewendet. Diese können über das **Dropdown-Menü Raum** ausgewählt werden. Vier dieser Referenzsysteme sind für den alltäglichen Gebrauch von praktischer Bedeutung:

- UV (2D) projiziert das Noise auf die UV-Koordinaten der Textur. Deformationen des Objekts werden berücksichtigt.
- Textur verwendet das im Textur-Tag angegebene Mapping. Deformationen des Objekts werden ignoriert.
- Objekt wendet das Noise dreidimensional auf das Achsensystem des Objekts an.
  Drehungen, Bewegungen und Skalierungen des Objekts werden berücksichtigt.
- Welt wendet das Noise dreidimensional auf das Achsensystem der Welt an, sodass das Noise an Ort und Stelle bleibt, während Drehungen, Bewegungen und Skalierungen des Objekts ignoriert werden.

In Bezug auf unser Thema, die Erzeugung volumetrischer Effekte, ist **Raum: Objekt** das Bezugssystem unserer Wahl, da in diesem Fall das Noise das Volumen des **D**P<sup>-</sup>

FOKUS

**3D & ANIMATION** 

DIGITAL ART

INTERACTIVE

Objekts dreidimensional durchdringt – oder das Volumen einer sichtbaren volumetrischen Lichtquelle. (Sie haben sich gerade "Aha!" gedacht, stimmts?)

#### Ebene- und Farbverlauf-Shader

Um komplexere volumetrische Strukturen zu schaffen, ist es wichtig, die Möglichkeit zu haben, verschiedene Shader innerhalb ein und desselben Materialkanals zu kombinieren, zu überlagern und zu maskieren. Zu diesem Zweck enthält das Cinema 4D den **Ebene-Shader**.

Um Shader in Cinema 4D modular und flexibel kombinieren zu können, ist der Ebene-Shader der Dreh- und Angelpunkt. Der Ebene-Shader fungiert als eine Art Container, in dem Ebenen (Shader oder Bitmaps) in einem Photoshop®-ähnlichen Stil gemischt, mit anderen maskiert oder in verschiedenen Ebenen-Modi verwendet werden. Die Ebenen-Modi beinhalten gängige Verfahren wie Normal, Multiplizieren und Ineinanderkopieren etc., aber auch einen Ebenen-Modus Maske. In diesem wird die darüberliegende Ebene mit den Graustufen der darunterliegenden Ebene maskiert.

Im Ebene-Shader können Bitmaps durch Anklicken des Buttons **Bild** geladen werden, Shader werden durch Klicken auf den Button **Shader** erzeugt. Mit einem Rechtsklick auf eine Ebene können Bitmaps und Shader auch kopiert (**Shader/Bild kopieren**) und eingefügt (**Shader/Bild einfügen**) werden.

Für eine bessere Organisation können Ordner per Klick auf den entsprechenden Button erzeugt werden. Zum Einfügen in einen Ordner wird eine Ebene auf einen Ordner gezogen, bis sich der Mauszeiger in einen kleinen vertikalen Pfeil ändert. Darüber hinaus können Sie mit dem Button Effekt Manipulationen wie Farbton, Sättigung, Kontrast, Distorter etc. auf der unter dem Effekt liegenden Ebene erzeugen.

Tipp: Wenn Sie einen Shader oder eine Bitmap in einen Materialkanal wie z.B. **Transparenz** laden und dann an derselben Stelle einen Ebene-Shader erstellen (durch Anklicken des kleinen Dreiecks rechts von "Textur"), dann ist der Shader oder die Bitmap automatisch als Ebene in den Ebene-Shader bewegt worden.

Maskieren ist ein wichtiges Thema innerhalb eines Ebene-Shaders, da Sie möglicherweise bestimmte Aspekte Ihres Shader-Setups auf bestimmte Bereiche Ihres Licht-Volumens beschränken möchten. Dazu sind Farbverlaufs-Shader geradezu ideal.

Anmerkung zum entsprechenden Node-Pendant: Die Entsprechung des Ebene-Shaders im Node-Materialsystem ist der **Ebene-Node**. Im Ebene-Node besteht



Bild 15 und 16 – Oben: Ein 3D-Noise (Sema) in einem scharfkantigen parallelen Spot mit sichtbarem volumetrischem Licht. Die Kanten wirken unorganisch scharf. Unten maskiert ein zylindrischer 3D-Farbverlauf den Sema innerhalb eines Ebene-Shaders und sorgt für weiche, aufgebrochene Kanten.

ebenfalls die Möglichkeit, Ebenen miteinander mit unterschiedlicher Opazität und Ebenen-Modus zu mischen. Jede Ebene bietet dabei einen offenen Port zum Einspeisen eines passenden Shader- oder Bild-Nodes. Ordner oder Ebenen-Effekte wie im Ebene-Shader entfallen. Ebenso gibt es keinen expliziten Ebenen-Modus zur Maskierung – die Maskierung einer Ebene geschieht vielmehr durch das Verbinden eines geeigneten Nodes mit dem Deckkraft-Port einer Ebene. Dieser wird im Ebene-Node per Rechtsklick auf die entsprechende Ebene erzeugt.

#### Farbverlauf-Shader

Der Farbverlauf-Shader erzeugt 2D-Farbverläufe auf Oberflächen oder 3D-Farbverläufe durch Objekte (oder Lichtvolumina) hindurch. Ein 2D-Farbverlauf kann linear entlang der U- oder V-Texturachse oder in bestimmten Formen wie Kreis, Box oder Stern verlaufen. Sie können per Winkel-Angabe gedreht, durch Turbulenz und Frequenz mit animierten Unregelmäßigkeiten aufgebrochen und durch Deaktivierung von Zyklisch unabhängig von Texturkacheln angewendet werden.

3D-Farbverläufe hingegen durchdringen das Volumen des Objekts oder eben der volumetrischen Lichtquelle. **Start-** und **Endwerte** von 3D-Farbverläufen beziehen sich auf das im Dropdown-Menü **Raum** ausgewählte Bezugssystem, z.B. das Achsensystem des **Objekts** oder der **Welt**. Ein Paar aus Start- und Endwert von -100 cm und 100 cm entlang der X-Achse (dem linken der drei Wertepaare bei Start und Ende) bedeuten bei einem linearen 3D-Farbverlauf im **Raum: Objekt**, dass dieser bei -100 cm auf der X-Achse des Objekts beginnt und bei 100 cm auf der X-Achse des Objekts endet. Startund Endwerte auf mehreren Achsen können verwendet werden, um einen linearen oder zylindrischen 3D-Farbverlauf anzuwinkeln.

Anmerkung zum entsprechenden Node-Pendant: Die Entsprechung des Farbverlaufs-Shaders im Node-Materialsystem ist der Node **Farbverlauf** oder **Basisfarbverlauf**. Beide Varianten unterscheiden sich lediglich in der Fülle und Detailtiefe der verfügbaren Parameter.

In Release 20 von Cinema 4D warten Farbverläufe mit einigen lang erwarteten Features zur interaktiven Bearbeitung auf:

- Ein Rechtsklick auf den Farbverlauf bietet nun die Option Größe zum Ändern der Größe der Darstellung.
- Sie können nun mit den altbekannten Hotkeys 1 und 2 durch den Farbverlauf pannen und zoomen, wie Sie es vom Cinema-4D-Viewport her kennen. Um die geänderte Ansicht zurückzusetzen, drücken Sie einfach H oder klicken Sie auf die kleinen schwarzen Klammern links und rechts vom Farbverlauf.
- Klicken Sie in den Verlauf, ziehen mit der Maus einen Rahmen und wählen so mehrere Knoten gleichzeitig aus.
- Sobald mehrere Knoten ausgewählt sind, können Sie mehrere Knoten oder mehrere Bias-Punkte zusammen verschieben,



Bild 17: Stellarer Nebel aus dem Titelbild dieses Artikels: Ein Material mit aktivem Transparenzkanal und einem Ebene-Shader sorgt für komplexe volumetrische Strukturen.

die gemeinsame Interpolation ändern und verschiedene Befehle durch Rechtsklick ausführen, z.B. **Knoten invertieren** etc.

#### Maskieren von Volumen-Noises

Und nun bringen wir alles zusammen: 3D-Farbverläufe sind ideale Werkzeuge, um 3D-Noises in einem Ebene-Shader zu maskieren. Lassen Sie uns ein einfaches Beispiel ansehen: In **Bild 15** wird das dreidimensionale Sema-Noise durch die Kanten des parallelen Spots (Radius 100 cm) exakt abgeschnitten beendet.

In **Bild 16** wird ein Farbverlauf im Modus **3D Zylindrisch** als Ebenenmaske verwendet. Der Gradient wird per Start- und Endwert entlang der Z-Achse der Lichtquelle angelegt. Damit hätten wir schon mal die Richtung des zylindrischen 3D-Verlaufs. Der **Radius** wird mit großzügigen 150 cm definiert, außerdem wird eine großflächige Turbulenz angewendet (bitte beachten Sie, dass eine große Turbulenzstruktur kleinere Werte benötigt und umgekehrt). Der so eingesetzte zylindrische 3D-Verlauf bricht die Kanten des 3D-Noise mit einer natürlich wirkenden Unregelmäßigkeit auf.

#### Fallstudie: Eine Frage der Zeit (Sternennebel)

Dieses Projekt für die TV-Dokureihe ZDF "Terra X" inszeniert den Vorgang der stellaren Kernfusion, die molekulare Struktur von Salz und das Prinzip der Uran-Blei-Datierung von Fossilien. Für die Sequenz über Wasserstoff- und Heliumatome in stellaren Nebeln habe ich genau das oben beschriebene Prinzip verwendet: eine (!) volumetrische Lichtquelle als Aquarium für miteinander gemischte und sich gegenseitig maskierende 3D-Shader. Das Ergebnis ist im **Titelbild** zu sehen.

Sehen wir uns diesen stellaren Nebel einmal von schräg oben an, fällt der ganze Schwindel auf (**Bild 17**): Die Kamera fährt linear durch ein sichtbares volumetrisches Licht eines parallelen Spots – sonst nichts. Ein hartes Turbulenz-Noise prägt im **Tab Noise** grob die großflächige Struktur des Lichts.

Im **Tab Sichtbarkeit** ist die **Helligkeit** des sichtbaren Lichts auf riesige 8000 % eingestellt, um auch dunklere Noise-Details darzustellen. Ein Material namens "Nebel" ist auf die Lichtquelle aufgebracht, welches nur einen aktiven Transparenzkanal mit einem integrierten Ebene-Shader enthält.

Innerhalb des Layer-Shaders sind 3D-Noises und 3D-Farbverläufe in Ordnern organisiert, die sich auf die logische Struktur des stellaren Nebels beziehen: Mega-Strukturen, mittlere Strukturen, kleine Strukturen ("Nebel Blau" etc...) Ordner werden wiederum wie normale Ebenen behandelt und durch Noises oder Farbverläufe maskiert.

Beim Ausklappen von Ordnern offenbaren diese ihren Inhalt: Noises unterschiedlicher Art werden durch einen Colorizer-Ebene-Effekt eingefärbt und durch 3D-Farbverläufe auf bestimmte Bereiche der Lichtquelle be-

#### Maxon Authorized Training Center (ATC)

Das Studio renderbaron ist Maxon Authorized Training Center (ATC) und bietet Schulungen zu Cinema 4D R2O 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 Cinema 4D. 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. 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.

schränkt/maskiert. Zur besseren Übersicht sind alle Layer entsprechend benannt.

Zusätzlich enthält die Szene ein Umgebungs-Objekt (Hauptmenü/Erzeugen/Umgebung), das per Checkbox **Nebel aktivieren** einen einfachen, aber effizienten schwarzen Nebel entlang der Z-Achse der Kamera erzeugt. Dadurch werden überflüssige optische Details in der Ferne ausgeblendet, die das Auge sonst nur ablenken würden.

#### Fazit

Mit diesem Artikel haben wir zwei Ansätze kennengelernt, wie man sichtbare oder sichtbare volumetrische Lichter zur Erzeugung von Wolken oder komplexen nebelartigen Strukturen verwendet. Im kommenden zweiten Artikel dieser Miniserie werden wir untersuchen, wie man eine animierte Sonnenkorona und sogar schattenwerfende fluffige Wolken erzeugt. **>ei** 



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 Cinema 4D 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.

www.renderbaron.de