

2018

ISSN 1433-2620 > B 43362 >> 22. Jahrgang >>> www.digitalproduction.com

Publiziert von DETAIL Business Information GmbH

Deutschland € 17,90

Österreich € 19,-

Schweiz sfr 23,-

2

DIGITAL PRODUCTION

DIGITAL PRODUCTION

MAGAZIN FÜR DIGITALE MEDIENPRODUKTION

MÄRZ | APRIL 02:2018



Landscapes

Digitale Landschaften für jede Produktion

Editing

Edius 9, Lesspain Kyno auf Windows und Avid First

Praxis

Maya 2018, Fusion, UE4, Modo Retopo und mehr ...



4 194336217907 02

Auf Distanz – Auch wenn der Name etwas anderes suggeriert, lassen sich die Animation Nodes auch für Standbilder einsetzen. In diesem Beispiel wurden sie dafür genutzt, die Objekte in der Vitrine anhand ihrer Entfernung zum Glas in der Farbe zu ändern.



Blender Animation Nodes 2.0

Im letzten Animation-Nodes-Artikel der Digital Production gab es einen kleinen Ausblick auf Version 2.0. Inzwischen ist diese erschienen und mit Blender 2.79 gibt es eine stabile Basis. Wer die Animation Nodes für Motion Graphics einsetzen will, kann sich freuen, dass mit dem neuen Falloff-Konzept viele Arbeiten jetzt intuitiver ausgeführt werden können. Außerdem wurde die Performance massiv gesteigert.

von Gottfried Hofmann

Möglich wurde die Leistungssteigerung, da die Animation Nodes unter der Haube jetzt nicht mehr in reinem Python, sondern in Cython geschrieben sind. Cython Code wird aber nicht mehr interpretiert, sondern kompiliert. Für den Anwender heißt das, dass man bei der Installation darauf achten muss, dass man die für das eigene Betriebssystem passende Installationsdatei verwendet. Herunterladen kann man sich die Pakete wie üblich auf https://github.com/JacquesLucke/animation_nodes/releases, dort stehen sie für Windows, Linux und Mac OS in 64-Bit-Ausführung zur Verfügung. Es geistern allerdings noch andere Kompilate durchs Netz.

Der Zuwachs bei der Performance kann sich durchaus sehen lassen. Nodes für ma-

thematische Operationen arbeiten jetzt ca. 200-mal so schnell und einen Spitzenplatz nimmt der Node zum Erstellen zufälliger Vektoren ein, der gegenüber Version 1.6 eine etwa 400-fache Geschwindigkeit aufweist.

Vektorisiert statt Schleifen

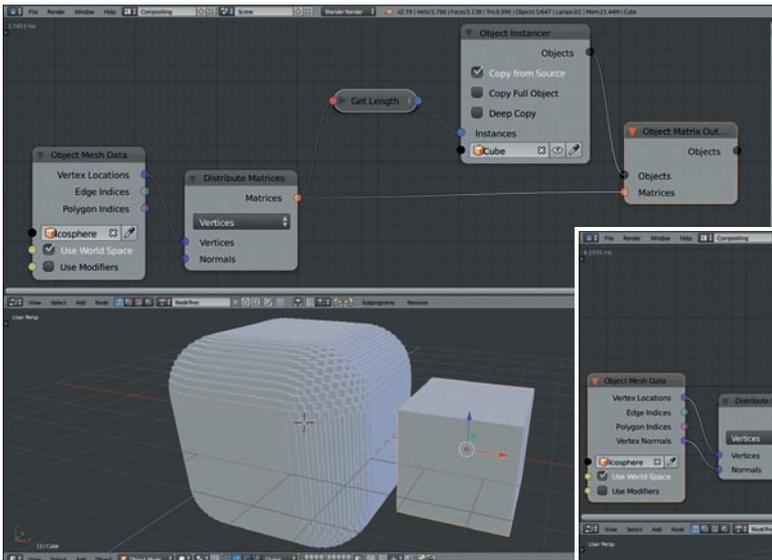
Wer sich bereits ein wenig mit den Animation Nodes auseinandergesetzt hat, der wird sich darüber freuen, dass in Version 2.0 viele Nodes und Datentypen vektorisiert wurden. Konkret bedeutet das, dass man mit diesen Nodes sowohl einzelne Datenpakete als auch ganze Listen verarbeiten kann, ohne dafür Schleifen bauen zu müssen. Weniger Schleifen heißt auch weniger Nodes und eine bessere Übersicht. Und da die Schleifen

nicht abgeschafft wurden, gibt es jetzt mehr Wege, einen gewünschten Effekt zu erreichen. Für einfache Aufgaben wird aber der Weg über vektorisierte Nodes empfohlen, da diese deutlich schneller arbeiten als manuell erstellte Schleifen.

Falloffs

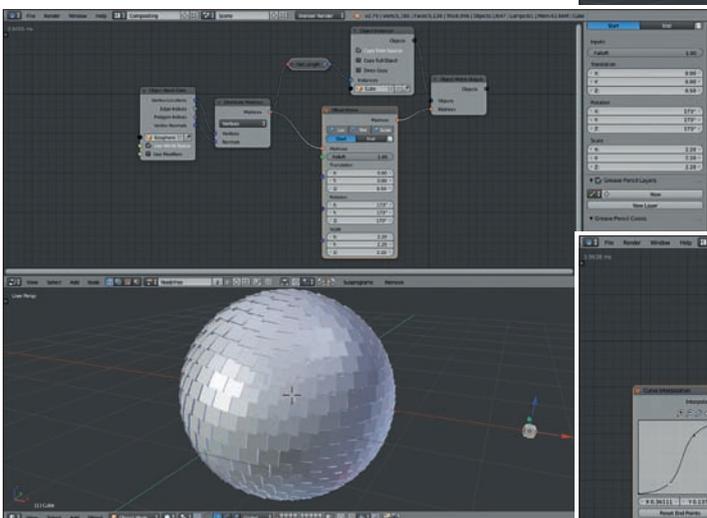
Ein weiteres neues Features in Animation Nodes 2.0 sind die sogenannten Falloffs. Diese funktionieren ähnlich wie die Effectors in Cinema 4D.

Ganz abstrakt gesprochen, lassen sich jetzt Einflussbereiche definieren, die die Stärke von Effekten steuern können. Modifizieren in Blender lassen sich auf ähnliche Weise über Vertex Groups steuern.



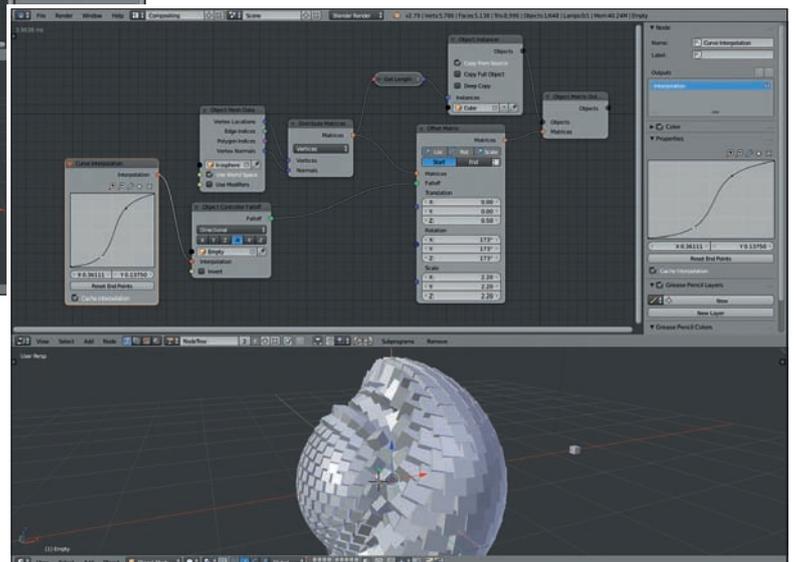
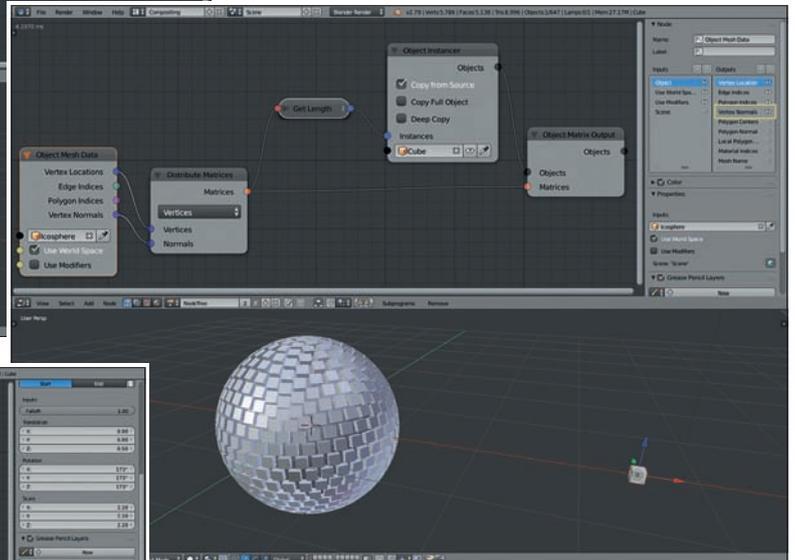
Links: Vorarbeit – Bis wir die Duplikate sehen können, müssen erst einmal vier Nodes hinzugefügt und richtig verschaltet werden (den Get Length Node fügt Animation Nodes automatisch hinzu). Nach dieser Vorarbeit kriegen wir aber alle Änderungen direkt im Viewport mit.

Unten: Versteckt – Der Ausgang für die Liste der Vertex Normals ist standardmäßig versteckt und muss erst aktiviert werden.



Oben: Schon wieder versteckt – Auch der Offset Matrix Node bietet versteckte Einstellungen. Grundsätzlich ist es bei den Animation Nodes immer eine gute Idee, rechts im Properties Panel nach zusätzlichen Optionen Ausschau zu halten.

Rechts: Den Falloff über ein Objekt zu steuern, ist sehr intuitiv. Für die Interpolation eignet sich der Curve Interpolation Node sehr gut.



Primitivlinge

Um das Beispiel möglichst einfach zu halten, verwenden wir nur zwei Basis-Objekte von Blender, sogenannte Primitives. Wir belassen den Default Cube und fügen eine Icosphere hinzu. An jedem Vertex der Icosphere wird später ein Cube platziert werden. Die Anzahl der Vertices bestimmt also, aus wie vielen Würfeln sich unser Objekt zusammensetzt.

Leider muss man diese Entscheidung bei den Primitives in Blender ganz am Anfang treffen und kann sie später nicht mehr ändern. In unserem Beispiel setzen wir die Subdivision auf 4. Dies kann man direkt nach dem Hinzufügen eines Objekts entweder in einem kleinen Sonderbereich im Tool Shelf ganz unten einstellen. Falls man das Tool

Shelf gerade nicht offen hat, kann man auch die Taste F6 drücken.

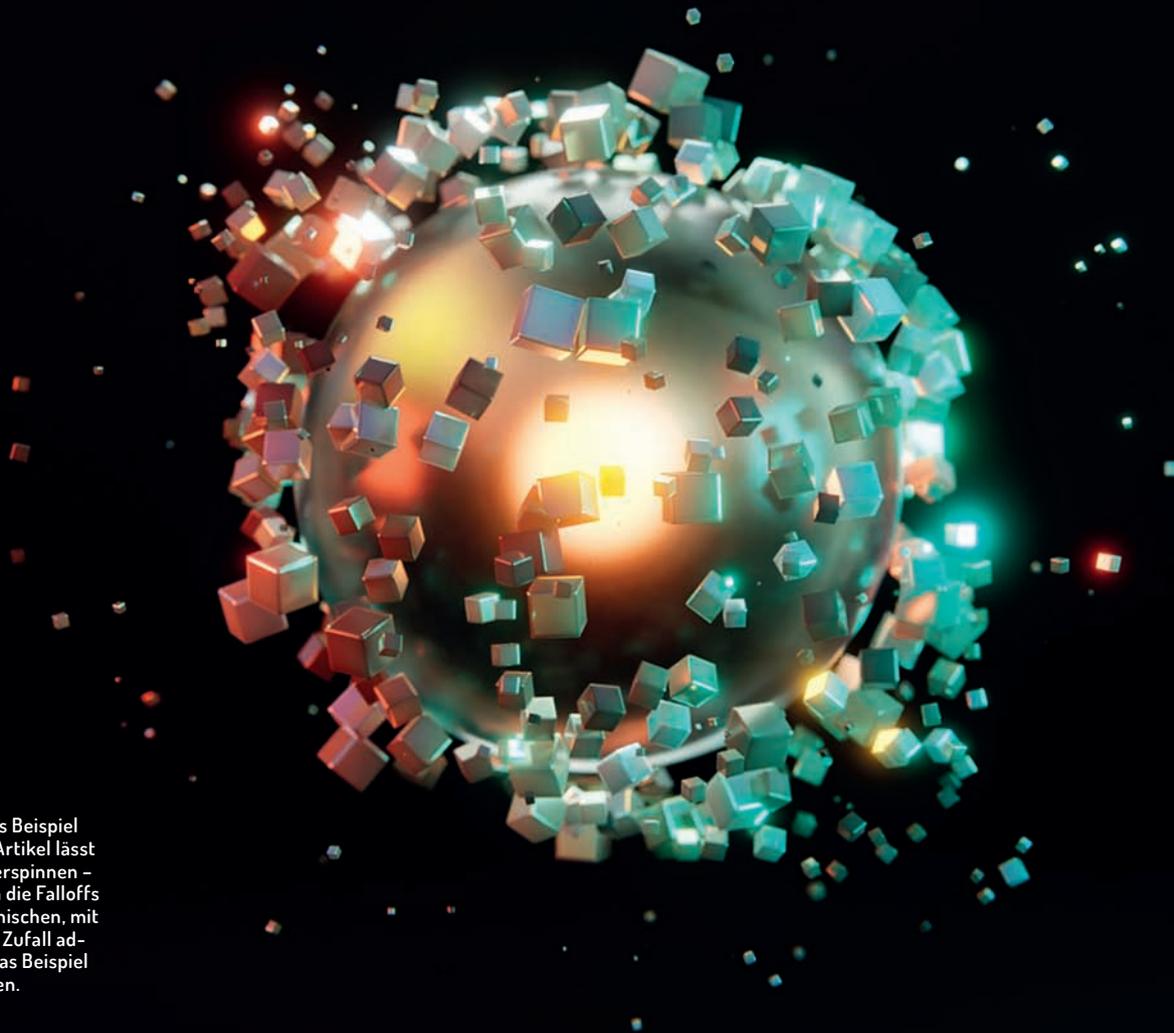
Unser erster vektorisierter Node

In diesem Beispiel können wir die Ergebnisse der ersten Schritte leider nur indirekt sehen. Erst nachdem wir vier Nodes zusammengesteckt haben, erscheinen die Früchte unserer Bemühungen. Zuerst fügen wir einen Object Instancer Node hinzu (Shift+A > Object > Instancer). Dieser erzeugt die Duplikate des Würfels. Im Objekt-Selektor des Nodes wählen wir den Cube.

Jeder duplizierte Cube soll transformiert werden. Dafür kommt der Object Matrix Output Node gerade recht (Shift+A > Object > Matrix Output). Die Object- und Matrix-

Eingänge dieses Nodes sind zu 25% transparent. Das bedeutet, dass man sowohl einzelne Objekte und Matrizen als auch Listen in den Eingang stecken kann. Mit anderen Worten: Er ist vektorisiert. Wir brauchen für diesen Node keinen Loop.

Der Ausgang des Object Instancer Nodes ist zu 50% transparent. Das bedeutet, es handelt sich hierbei um eine Liste. Wenn wir ihn mit dem Object-Eingang des Object Matrix Output Nodes verbinden, so wird auch der Ausgang des besagten Nodes zu 50% transparent, sprich die Ausgabe ist wieder eine Liste. Solche Kleinigkeiten im Interface der Animation Nodes sind unauffällig, aber äußerst hilfreich, und zeigen, dass sich der Entwickler viele Gedanken über Usability gemacht hat.



Zufall: Das Beispiel aus dem Artikel lässt sich weiterspinnen – man kann die Falloffs nämlich mischen, mit ein wenig Zufall addiert ist das Beispiel entstanden.

Matrizen sind einfach besser

Warum verwenden wir den Object Matrix Output Node statt des Object Transform Output Nodes wie im Coin-Stack-Beispiel des letzten Animation-Nodes-Artikels? Nun, Matrizen kodieren gleich drei Transformationen auf einmal, nämlich Translation, Rotation und Skalierung. Wir sparen uns dadurch ein paar Nodes, was auch der Übersicht zugutekommt. Und wir können einen Node einsetzen, der die benötigten Matrizen direkt aus einer Liste von Vertices und Normalen ausliest: Distribute Matrices (Shift+A > Matrix > Distribute).

Den Ausgang des Distribute Matrices Nodes verbinden wir mit dem Matrix-Eingang des Object Matrix Output Nodes und mit dem Instances-Eingang des Object Instancer Nodes. Dabei fügt Animation Nodes automatisch einen Get List Length Node als Konverter zwischen. Die Anzahl der Würfel-Duplikate ist jetzt immer gleich der Anzahl der Matrizen aus dem Distribute Matrices Node.

Im Dropdown-Menü des Distribute Matrices Nodes wählen wir „Vertices“. Als Eingänge habe wir nun eine Liste von Vertices und

eine Liste von Normalen. Diese können wir über einen Object Mesh Data Node beziehen (Shift+A > Mesh > Object Mesh Data). Dort setzen wir die Icosphere als Quelle und verbinden den Vertices-Ausgang mit dem Vertices-Eingang des Distribute Matrices Nodes. An dieser Stelle sollten die Duplikate des Würfels erscheinen, allerdings sieht das Ganze noch nicht aus wie gewünscht, da die Würfel zum einen viel zu groß und zum anderen alle gleich rotiert sind. Dadurch entsteht eine Art Würfel mit abgerundeten Kanten anstatt der gewünschten Kugel aus Würfeln.

Gut versteckt

Die Größe der Würfel können wir sehr gut kontrollieren, indem wir den ursprünglichen Würfel im Edit Mode skalieren. Die Ausrichtung erreichen wir über die Vertex Normals. Deren Ausgang ist beim Object Mesh Data Node standardmäßig versteckt. Wenn wir den Node anwählen, sehen wir rechts im Properties Panel die Inputs und Outputs. Die Liste der möglichen Ausgabe-Sockets ist ziemlich lang – verständlich, dass standardmäßig nicht alle davon sichtbar sind. Ein

Klick auf das Augen-Symbol Vertex Normals lässt den entsprechenden Ausgang erscheinen, und wir können ihn mit dem Normals-Eingang des Distribute Matrices Nodes verbinden. Jetzt sind alle Würfel entsprechend der Vertex-Normalen ausgerichtet, sprich sie zeigen von der Oberfläche der Kugel weg.

Schiebung

Jetzt ist der Punkt erreicht, an dem die Falloffs ins Spiel kommen. Falloffs können in Animation Nodes nicht mit jedem Node eingesetzt werden, sondern nur mit denjenigen, die die entsprechenden grünen Eingänge mitbringen. Ein solcher Node ist der Offset Matrix Node (Shift+A > Matrix > Offset). Indem wir ihn über die bestehende Verbindung zwischen Distribute Matrices und Object Matrix Output ziehen, schaltet sie sich automatisch dazwischen.

Im rohen Zustand ist dieser Node reichlich unscheinbar. Erst wenn man im oberen Bereich die Transformationen aktiviert, die man manipulieren will, zeigt sich sein Funktionsumfang. Wir selektieren sowohl Loc, Rot als auch Scale, und es erscheinen die entsprechenden Eingänge. Wenn man jetzt

einen Wert verändert, wird diese Änderung auf alle Würfel auf einmal übertragen.

Die Grundidee des Beispiel-Effekts ist ein Ausgangs- und ein Endzustand, zwischen denen mittels Falloff interpoliert wird. Den Endzustand haben wir bereits in Form der Würfel-Kugel, die wir gebaut haben. Den Ausgangszustand definieren wir jetzt in diesem Node. Für das Beispiel ist das eine Scale von 2,2 auf allen Achsen, eine Rotation von 173 Grad auf allen Achsen und eine Translation von 0,5 auf der X-Achse.

Besser versteckt

Die Translation passiert standardmäßig allerdings global. Auch hier bietet Animation Nodes mal wieder eine versteckte Option. Mit selektiertem Node finden sich im Properties Panel ganz unten die Advanced Node Settings, wo wir die Translation auf „Local Axis“ umstellen.

Im Allgemeinen lohnt es sich, bei jedem Node nachzusehen, ob es nicht erweiterte Einstellungen gibt. Das geht besonders einfach mit der Taste „U“. Und natürlich sollte man immer die offizielle Dokumentation zu Rate ziehen (<http://animation-nodes-manual.readthedocs.io/en/latest/>).

Herabfallen

Falloffs sind in Animation Nodes ein eigener Datentyp, der die neuen grünen Sockets verwendet. Natürlich gibt es dafür auch eine Reihe an Nodes. Für unser Beispiel ist der Object Controller Falloff gut geeignet, weil sich dieser über ein Objekt kontrollieren lässt (Shift+A > Falloff > Object Controller). Als Objekt für den Controller fügen wir ein Empty zur Szene hinzu. Damit es später durch die Würfel-Kugel sichtbar ist, vergrößern wir die Display Size des Empties auf 2. Außerdem setzen wir es als Objekt in den Object Controller Falloff Node und verbinden die beiden grünen Sockets.

Wenn wir jetzt das Empty in die Nähe der Würfel-Kugel bewegen, sehen wir schon einen Effekt. Es ist aber noch nicht ganz das, was erwünscht ist, denn der Falloff Type ist noch auf „Sphere“ eingestellt. Ein Wechsel auf „Directional“ zeigt einen Effekt, der näher am gewünschten Ergebnis ist. Jetzt funktioniert der Falloff nicht mehr punktförmig vom Zentrum des Empties aus, sondern von einer unendlich großen, unsichtbaren Fläche. Je nach Richtung, in die das Empty später bewegt wird, sollte allerdings noch die Achse geändert werden. Für eine Bewegung von Links nach Rechts entlang der X-Achse z.B. auf „-X“.

Hier eröffnet sich eine Reihe von Möglichkeiten. Z.B. können wir die Breite des Falloffs direkt kontrollieren, indem wir das Empty größer oder kleiner skalieren. Wie der Übergang im Detail aussieht, lässt sich über die Interpolation regeln. Hier können wir entweder eines der viele Presets wie z.B. quadratisch wählen oder die Interpolation völlig frei über eine Kurve definieren (Shift+A > Interpolation > From Curve). >ei



Gottfried Hofmann ist Diplom-Informatiker und bietet seit mehreren Jahren professionellen Support sowie Schulungen für die freie 3D-Software Blender an. Als freischaffender Autor schreibt er für Fach- und Computerzeitschriften. Er hat zahlreiche Blender-Tutorials verfasst, u.a. für CG Tuts+ und CG Cookie. Weiterhin betreibt er die Webseite www.Blender-Diplom.com, auf der Blender-Tutorials in deutscher und englischer Sprache zur Verfügung stehen und Schulungen gebucht werden können.

Anzeige

JETZT KENNENLERNEN

DIGITAL PRODUCTION



FACHMAGAZIN FÜR DAS GESAMTE
SPEKTRUM DIGITALER MEDIENPRODUKTION

VISUAL EFFECTS
ANIMATION
VISUALIZATION & DESIGN
GAME & INTERACTIVE
SCIENCE & EDUCATION

www.digitalproduction.com/2for1

Ihre Service-Hotline
T: +49(0)61 23 92 38-2 13
F: +49(0)61 23 92 38-2 14
E: order@digitalproduction.com