

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,-

5

DIGITAL PRODUCTION

# DIGITAL PRODUCTION

MAGAZIN FÜR DIGITALE MEDIENPRODUKTION

SEPTEMBER | OKTOBER 05:2018



## Workstations

Elf Maschinen im Härtestest  
- was taugt für Artists?

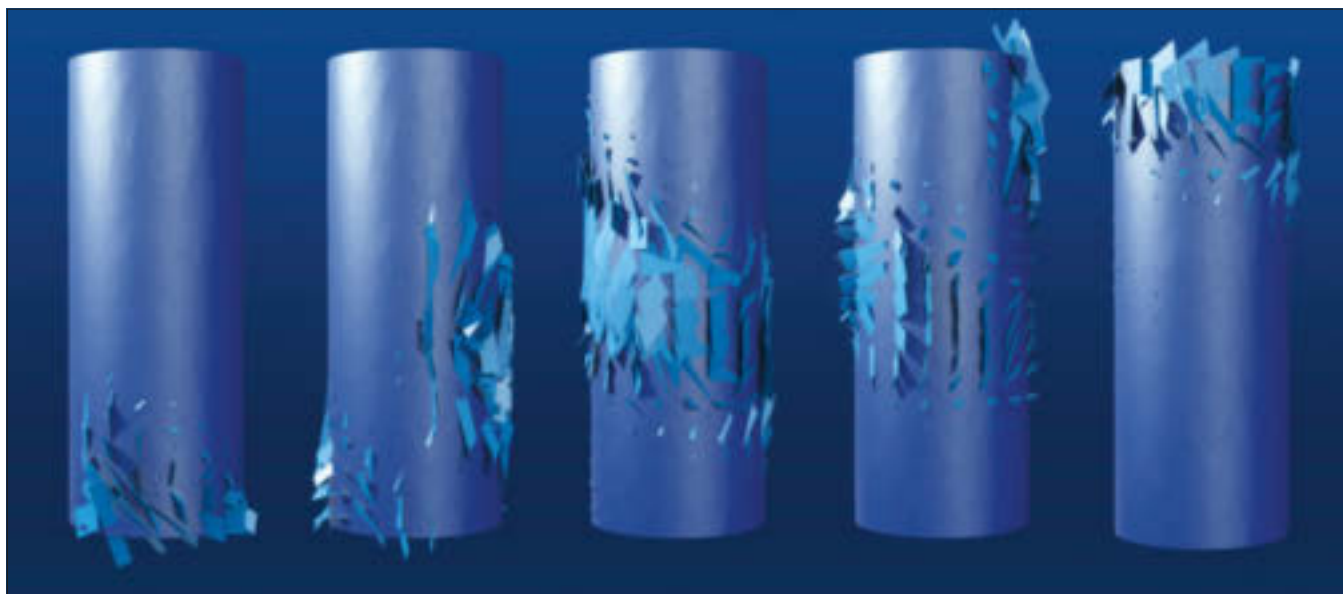
## Neue Tools!

Cinema 4D R20, Fairlight,  
Nuendo & CineXtools

## und vieles mehr!

Flame, Modo, Pro Render,  
FX Prototyping, Vue ...





## Smartes Rapid FX Prototyping in 3ds Max 2019

Aufwendige Feature Shows sind geschmückt mit imposanten visuellen Effekten, um das digitale Set oder die Fähigkeiten einzelner Schauspieler gekonnt in Szene zu setzen. Als Beispiel dient eine Szene aus dem X-Men-Universum, in der ein junges Mädchen namens Raven Darkholme – später bekannt als Mystique – ihr äußeres Erscheinungsbild verändert. Ihr natürliches Aussehen besteht aus leuchtend roten Haaren, gelben Augen ähnlich einem Reptil und einer Hautoberfläche in dunklen Blautönen, die teilweise durch Muster verziert ist. Ist man Benutzer von 3ds Max 2019, dann kann man mit Zuhilfenahme des Max Creation Graph nebst dem Data Channel Effekte für die Formwandlung aufbauen sowie anpassen.

von Rainer Duda

Die fiktive Figur namens Mystique, gespielt von Rebecca Romijn und Jennifer Lawrence in „X-Men: Apokalypse“, kann ihre natürliche Form anpassen. Im Feature nimmt die Mutantin Mystique die Form unterschiedlicher Personen an. Während dieses Formwandlungsprozesses wird die ursprüngliche reptilienähnliche blaue Haut, gezielter einzelne Schuppen partiell deformiert. Fast als würden sich die Schuppen samt einer imposanten Animation umstülpen und eine andere Form darstellen. Und diese Deformation findet am kompletten Körper statt, aber nicht zur gleichen Zeit, sondern partiell. Um diesen Effekt in der aktuellsten Version von 3ds Max nachzubilden, muss der gesamte Effekt in einzelne Komponenten zerlegt und diese der Reihe nach aufgebaut werden. Der modulare Aufbau des Effekts ermöglicht die schrittweise Verfeinerung im späteren Verlauf und ermöglicht den Aufbau des Effekts in unterschiedlichen Ebenen.

### Erste Nachbildung der Schuppen

Bevor überhaupt an den Effekt an sich gedacht wird, muss die Grundlage in Form von Schuppen aufgebaut werden. Der Einfachheit halber nutzen wir an erster Stelle



Mehr Einfluss auf Objekte  
durch Data Channel Modifier

einen Zylinder bestehend aus Polygonen, dessen Deckel und Boden entfernt wurden. Wichtig ist für den ersten Test, dass die Auflösung der Polygone ein Muster aus viereckigen Polygonen ergibt. Wichtig ist, dass jedes einzelne Polygon nun als eigenständiges Element vorhanden ist. Wobei eigenständig nicht bedeutet, als separates Objekt in der Hierarchie geführt zu werden, sondern die einzelnen Polygone zu schneiden, damit sie keine Verbindung zu den anderen haben.

Nun ist es so, dass die standardmäßigen Funktionen und Modifier nicht den gewünschten Zweck erfüllen und digitale Künstler\*innen selbst eine Lösung erstellen wollen. Für diesen Fall bietet sich der Max Script Creation Graph wunderbar an – kurz MCG. Dahinter verbirgt sich quasi visuelles Max Scripting für die Erstellung eigener Modifier oder Objekte für das Create Panel. Erstellen lässt sich ein MCG über das Scripting-Menü und einen Klick auf die Schaltfläche „New Max Creation Graph“.

Ein neues Fenster öffnet sich mit einem Graphen in der Mitte, links die verfügbaren Nodes und rechts deren Eigenschaften und darunter ein Log-Ausgabe-Bereich. Zunächst geht es darum, die Basis zu definie-

ren. In diesem Fall soll ein neuer Modifier entwickelt werden, der auf alle Objekt angewendet werden kann und die einzelnen Flächen in separate Häppchen zerschneidet.

Damit Nodes einfacher gefunden werden können, lässt sich die Taste X betätigen. In dem Suchfenster kann mit Schlagworten nach speziellen Nodes gesucht werden. Um nun die Entwicklung eines Modifiers zu starten, müssen zwei Nodes in den Graphen eingefügt werden: einmal Modifier Mesh, ein Node, der den Input der Geometrie im Stack einliest, und ein Modifier Output Node. Dazwischen wird nachfolgend die Logik aufgebaut. Es empfiehlt sich, das Tool bereits in der frühen Phase abzuspeichern und einen passenden Namen zu vergeben, über den das Tool in der Modifier-Liste gefunden und angewendet werden kann. Über das Tastenkürzel Strg+E wird der Node-Baum evaluiert und auf Fehler überprüft.

Spannend für den Ausbau des Graphen ist, dass Autodesk bereits oft benötigte Funktionen in Nodes aufbereitet hat. Somit lässt sich nun direkt der Node namens „Split Mesh into Polygons“ erstellen und an den Mesh Input Node anknüpfen. Würde man den neu erstellten Node nun mit dem Output Node verknüpfen wollen, dann wird dies nicht funktionieren, da ein Mesh als Ausgabe erwartet wird und kein Array mit Objekten. Daher müsste nun noch ein Attach all Meshes Node erstellt und dazwischen eingefügt werden. Alle einzelnen Flächen werden somit zu einem Objekt (Mesh). Die Eingänge und Ausgänge der Nodes sind ebenfalls farblich kodiert und mit unterschiedlichen Symbolen ausgestattet, um die unterschiedlichen Funktionen und Datentypen einfacher ersichtlich zu gestalten. Prinzipiell würde das System nun funktionieren. Wichtig ist, dass nach jeder Änderung gespeichert und evaluiert wird.

### Data Channels for President!

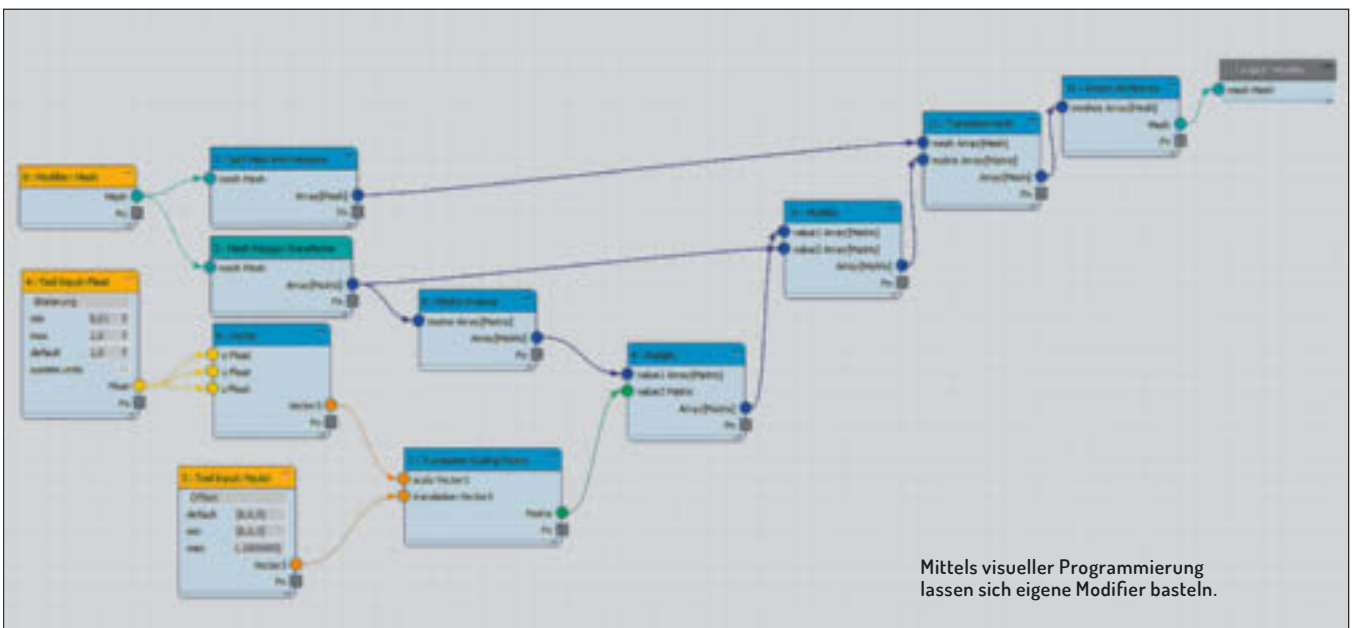
Der Data Channel Modifier ist für viele 3ds-Max-Nutzer ein wahrer Segen – ein Werkzeug, das die Automatisierung komplexer Modellierungen möglich macht. Doch was kann man sich darunter vorstellen? Objekte bestehen aus einer Vielzahl unterschiedlicher Sub-Kategorien von Daten wie unterschiedliche Geometrie-Ebenen (Vertex, Edge, Face), Vertex Colors, Welt-Koordinaten, bestimmte Selektionen (Volume Selection), Textur-Mapping-Informationen und vielen weiteren. Durch den Data Channel Modifier wird ein Operator-Stack bereitgestellt, der es erlaubt, die Daten individuell abzugreifen, mit Funktionen zu belegen und auszugeben. In diesem Beispiel wird die Hauttransformation durch Nutzung des Data Channel Modifiers realisiert.

Der erste Schritt besteht jedoch darin, dem zugrunde liegenden Zylinder zunächst den Modifier zuzuweisen, der zuvor erstellt wurde – die Trennung einzelner Flächen. Daraufhin wird der Data Channel Modifier hinzugefügt. Im Feld „Parameters“ befindet sich der zuvor erwähnte Operator-Stack. Bevor der Operator-Stack befüllt wird, muss zuvor noch ein kleines Setup in der Szene aufgebaut werden. Die Oberfläche des Zylinders stellt die Haut dar. Die soll aber nicht auf einmal transformiert werden, sondern partiell wie in der Feature Show „X-Men“. Also wird zunächst eine Spirale um den Zylinder erstellt durch die Nutzung einer Helix im Create Shapes Panel. Die Spirale muss tiefer gelegen sein als der Boden des Zylinders und höher enden als die Spitze des Zylinders. Des Weiteren wird noch der Initiator des Effekts benötigt. Hierzu dient ein Punkt oder auch Null-Objekt, das bei den Helper-Objekten zu finden ist. Dem Null-Objekt wird ein Motion

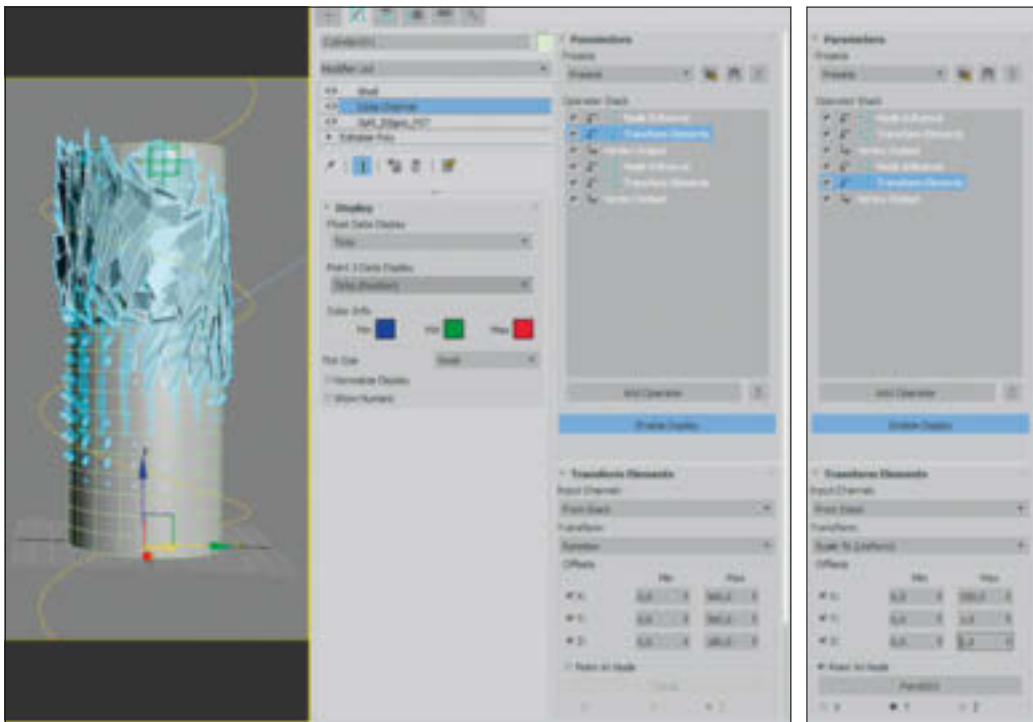
Controller für die Position zugewiesen – zu finden im Motion Panel über das Plus-Symbol, sobald die Positionszeile selektiert wurde. In den Auswahlmöglichkeiten des Motion Controllers wird auf die Option „Path Constraint“ geklickt. Nun muss die Spirale als Pfad festgelegt werden. Der Helfer folgt automatisch dem Pfad von Anfang bis Ende basierend auf den zugrunde liegenden Frames in der Zeitleiste. Während man den Regler der Zeitleiste verschiebt, wird die Position des Helfers verändert. An dieser Stelle kann das Motion Panel verlassen und das Modifier Panel geöffnet werden. Nun muss der Data Channel Modifier des Zylinders erneut in den Fokus geholt werden.

### Transformation in Gang gesetzt

Der Datenfluss im Operator-Stack ist durch einen Input Operator definiert, danach folgt der Process Operator, und es muss ein Output Operator danach eingebunden werden. Was möchte ich als Input definieren? Wie soll der Input verarbeitet werden? Was soll nach der Verarbeitung ausgegeben werden? Drei Fragen, die den Fluss vereinfacht beschreiben. Die Funktion „Add Operator“ unterhalb des Operator-Stacks öffnet die Liste mit allen verfügbaren Operatoren. Der erste benötigte Operator nennt sich „Node Influence“ und wird per Drag-and-drop in den Operator-Stack verschoben. Die Einstellungsmöglichkeiten der Operatoren werden nach der Selektion direkt unterhalb des Operator-Stacks angezeigt. Als erstes muss der Helfer als treibender Node beim Influence Operator hinterlegt werden, gefolgt von der Einstellung, dass die Influence auf die Elemente wirken soll. Die Hold Time kann vorab auf einen Wert von 70 gestellt werden. Ein Parameter, der angibt, wie lang der



Mittels visueller Programmierung lassen sich eigene Modifier basteln.



Es können mehrere Datenflüsse erstellt werden, die sich gesondert um die Steuerung von Skalierung sowie Rotation kümmern.

Effekt auf den vorangegangenen Elementen nachwirken soll, bis er in den Ausgangszustand zurückfällt. Der maximale Radius des Influence Operators muss nach eigenem Bedarf angepasst werden. Hierbei kommt es auf die System-Units an und wie groß der Zylinder ist sowie wie weit die Spirale (Helix) vom Zylinder entfernt ist. Im Beispiel wird ein Wert von 40 genutzt.

Als zweiter Operator wird nun ein Transform Elements Operator hinzugefügt. Der Operator an sich ist selbsterklärend, nicht aber, wie er eingestellt werden muss. Von Seiten Autodesk wird empfohlen, den Modus des Operators grundsätzlich auf Replace zu setzen. Das geschieht über die rechte Maustaste auf den Operator und die Auswahl „Replace“. Die Änderung wird sichtbar, indem das Symbol des Operators in ein Input-Symbol geändert ist. Als Transformationsmethode für die Elemente der zugrunde liegenden Objekte wird als nächster Schritt „Rotation“ ausgewählt. Wichtig ist, die Offsets für X, Y und Z zu aktivieren und die Werte X und Y auf 360 zu setzen, während Z auf 180 geändert wird. Das Verhalten soll beispielhaft am Zylinder verdeutlichen, dass die Schuppen sich unterschiedlich stark transformieren, indem für das Auge deutlich mehr visuelle Energie an den Transformationsbereichen erzeugt wird.

Um den Effekt nun im Viewport und für das Rendering sichtbar zu gestalten, muss als nächstes ein Vertex Output Operator in den Operator-Stack eingefügt werden. Nun ist das visuelle Ergebnis bis zu diesem Schritt zu sehen. Die Farbkodierungen im Viewport lassen sich innerhalb des Display-Feldes des Data Channel Modifiers anpassen.

## Skalierung anpassen

Der Prototyp des Effekts wird nachfolgend mit einer Funktionalität ausgestattet, um die Skalierung der Flächen zu vergrößern, je näher der Helfer an die Geometrie des Zylinders gelangt. Hierfür wird ein weiterer Datenfluss in den Operator-Stack eingebaut, direkt unterhalb des bestehenden Flows. Besser gesagt: Es muss ein Flow aufgebaut werden, der dem ersten quasi gleicht. Es werden somit drei Operatoren benötigt: erstens ein Node Influence Operator. Dazu gesellt sich ein Transform Elements Operator im Replace Mode sowie ein Vertex Output Node, um den Datenfluss zu vollenden. Wichtig ist, dass neben den identischen Einstellungen für den Node Influence Operator Anpassungen im Bereich der Transform Elements durchgeführt werden. Anstelle der Rotation soll diesmal rein die Skalierung (Uniform) beeinflusst werden. Das Minimum kann bei dem Wert Null bleiben, jedoch der Maximalwert sollte auf mindestens 150, besser 200 angehoben werden. Um dem Effekt weitaus mehr visuelle Energie für das Auge des Betrachters zu liefern, wird die Funktion „Point at Node“ aktiviert und der Helfer als Node angefügt nebst der aktiven Funktion „Individual Elements“.

Die bis jetzt angeführten Einstellungen lassen das Helfer-Objekt um den Zylinder spiralförmig kreisen. Je näher das Objekt an den Zylinder kommt, desto größer werden die symbolischen Schuppen skaliert, in drei Dimensionen rotiert und orientiert gemäß der Position des Helfers. Bis hierhin sollte der Effekt eine plausible visuelle Energie erzeugen. Der Effekt lässt sich in Szene setzen,

indem man das Grundobjekt des Zylinders dupliziert und 0,1% in X und Y nach innen skaliert. So wird die Anpassung des Prototypen-Effekts vereinfacht.

## Weiterführende Gedanken

Noch bewegt man sich in der aktiven Entwicklung eines Prototypen für den gewünschten Transformationseffekt wie bei Mystique. Entwickler\*innen können nun in Betracht ziehen, die Skalierung und ein zusätzliches Offset der Schuppen zu realisieren. Der Flexibilität halber werden die Gedanken in den MCG eingebracht. Ausgangslage ist der bereits bestehende Graph, der Geometrie entgegennimmt und in einzelne Flächen zerschneidet.

Als erstes werden zwei Parameter benötigt, die nach außen hin einstellbar sein sollen. Dafür werden zwei Tool Input Nodes erstellt: ein Float sowie eine Vector-Variante. Der Float Input wird mit „Skalierung“ betitelt, der Vector bekommt das Label „Offset“. Bei der Skalierung wird der Wertebereich 0 bis 1 definiert mit einem Standardwert von 1. Beim Offset hingegen wird das Minimum auf 0 gesetzt, während das Maximum auf dem Standard belassen werden kann. Der Default-Wert beim Offset wird 0 für X, Y und Z. Da es sich um eine gleichmäßige Skalierung in alle Dimensionen handeln soll, muss ein Vector Node erstellt werden, dessen drei Inputs mit dem Output des Skalierungs-Inputs verbunden werden. Vorrätig ist nun eine Skalierung sowie eine Positionsveränderung. Um beide Parameter zu verbinden wird demnach eine Translation-Scaling-Matrix benötigt. Der Vector Node der Skalierung wird mit dem Scale Input des Matrix

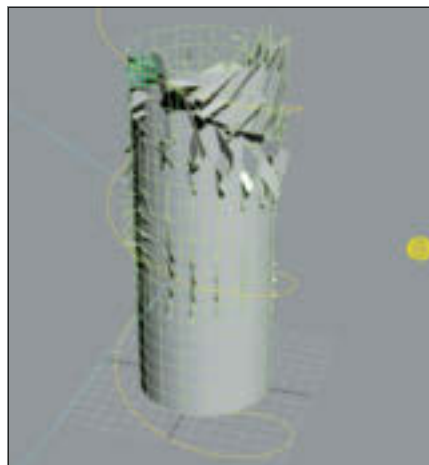
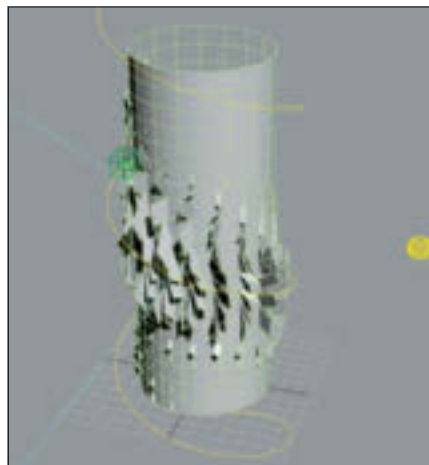
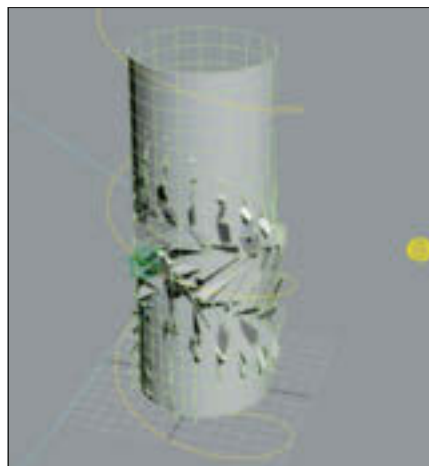
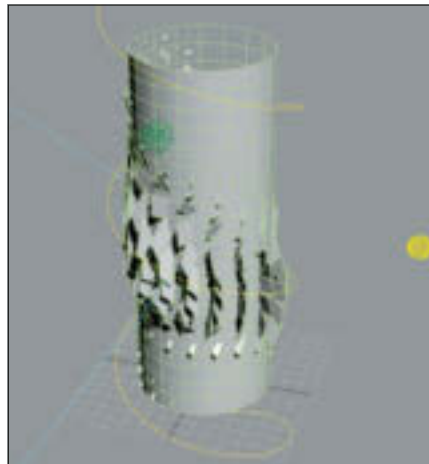
Nodes verbunden. Der noch offene Input der Translation kommt vom Offset Vector. Bevor es an dieser Stelle weitergeht, müssen die Transformationen der jeweiligen Polygone des zugrunde liegenden Objekts bereitgestellt werden. Das geschieht über den Node namens „Mesh Polygon Transforms“, dessen Input mit dem Output des Modifier Mesh Input Nodes verbunden wird.

Bevor nun Skalierung und Offset stattfinden, muss ein bestimmtes Prinzip realisiert werden. Nämlich müssen die einzelnen Polygone auf den lokalen Mittelpunkt (Origin) ausgerichtet werden. Bildlich kann man sich vorstellen, dass die individuellen Polygone auf dem Nullpunkt liegen mit einem zentrierten Pivot-Element. Daraufhin werden Skalierung und Offset angewendet, danach werden die Polygone zurück an ihre ursprüngliche Position gebracht.

Für die Realisierung des ersten Schritts, der Platzierung der Polygone am Local Origin, wird ein Matrix Inverse Node erstellt und mit dem Input des Mesh Polygon Transform Nodes verknüpft. Die Anwendung von Offset und Skalierung wird durch die Multiplikation vom Matrix Inverse Node sowie der Translation Scaling Matrix durchgeführt. Wichtig zu wissen ist, dass der erste Input des Multiply Nodes das Ausgabeformat definiert und mit Vorsicht gearbeitet werden muss. Der erste Input des Multiply Nodes ist somit ein Array mit Polygonen und der zweite Input die Matrizen (Offset und Skalierung).

Um nun nach getaner Arbeit, nach der Anwendung von Offset und Skalierung die jeweiligen Polygone wieder an ihre ursprüngliche Position zu bringen, muss erneut ein Multiply Node erstellt werden. Der erste Input des neuen Multiply Nodes wird mit dem Output des vorigen Multiply Nodes verbunden, also dem Ergebnis, wenn die Polygone basierend auf ihrem lokalen Mittelpunkt transformiert werden. Der zweite Input wird mit dem Output des Mesh Polygon Transform Nodes verknüpft. Durch die Multiplikation werden nun die transformierten Polygone wieder an ihre ursprüngliche Position gebracht.

Nun müssen die Ergebnisse der zweiten Multiplikation auf die unterschiedlichen Polygone übertragen werden. Dies geschieht in zwei Stufen. Für die erste Stufe muss ein Node mit dem Namen „Transform Mesh“ erstellt werden. Der Node nimmt ein Array an Elementen entgegen und transformiert die Elemente entsprechend dem zugrunde liegenden Array an Matrizen. Um nun auch den benötigten Output für den Modifier zu erhalten, nämlich ein Mesh, wurde zuvor der Attach All Meshes Node erstellt. Der Output des Transform Mesh Nodes muss nun an den Attach All Meshes Node angehängt werden. An dieser Stelle muss der gesamte Code evaluiert und gespeichert werden.



Effektprüfung im Viewport

Prüft man daraufhin den Modifier, dann werden die neuen Parameter sichtbar.

## Verbindungen schließen

Die beiden neu gewonnenen Parameter beeinflussen den Effekt positiv. Werden an dieser Stelle weiterführende Hilfsobjekte eingefügt, dann lassen sich einzelne Parameter auch mit anderen Parametern verknüpfen. Die neu gewonnene Verbindung erlaubt die Steuerung von Animationen und Simulationen auf eine automatische Art und Weise. Der Einfachheit halber empfiehlt sich die Tastenkombination Alt+5. Alternativ lässt sich auch ein Objekt selektieren, dessen Attribute man mit einem anderen Objekt verknüpfen möchte. Ist ein Objekt selektiert, dann muss im Menü „Animation“ nach „Wire Parameters“ gesucht werden. Das kleine Menü lässt den Anwender entweder das zuvor erwähnte Fenster direkt öffnen oder die Funktion „Wire Parameters“ ausführen. Letztere erlaubt dem Anwender, ein Zielobjekt zu selektieren und auf dem Objekt direkt einen Parameter auszuwählen. Quasi eine geführte Anwendung anstatt eines Fensters, bei dem manuell Einstellungen durchgeführt werden müssen. Kurz: Die lange Suche bleibt erspart. Wurde das Paar an Parametern ausgewählt, dann muss nur noch über den Befehl „Connect“ eine Verbindung hergestellt werden.

Zieht man nun die in dieser Publikation aufgezeigten Möglichkeiten des Prototypings bei der Entwicklung von Effekten in Betracht, dann wird ein nahezu unerschöpfliches Potenzial freigesetzt. Mit einem Schlag ist man nicht mehr abhängig von vorhandenen Tools mit teilweise rudimentärer Funktionalität und schreibt seine eigenen. Reicht das Mesher-Objekt in den Compound-Objekten nicht aus, da die Position des Partikelsystems, aus dem man ein Objekt generieren möchte, nicht vom Mesher in Betracht gezogen wird, da er sein eigenes nimmt, dann lässt sich mittels MCG in Windeseile ein eigenes Objekt entwickeln, das weitaus mehr Funktionalität als der Mesher bereitstellt. Zum Beispiel das manuelle Hinzufügen von Punkten im künstlerischen Kontext, um weitere Objekte in der Szene zielgerichtet zu platzieren.

Auch wenn die Graphen bei der einen oder anderen Anwendung sehr komplex ausfallen, Autodesk hat im Gegensatz zum Slate Material Editor beim MCG die Möglichkeit offen gelassen, mit Compounds zu arbeiten. Sozusagen Sub-Netzwerke eines bestehenden komplexen Graphen, eine Reihe an MCG Nodes, die mit Compound-Inputs und Outputs versehen werden, um sie daraufhin als eigenständige Operatoren in der Node-Liste abzulegen. Die neu erzeugten Nodes können wie selbstgestellte MCG Modifier sowie MCG-Objekte innerhalb eines Produktionsteams ausgetauscht werden. > ei