

Cinema 4D und Flash

Viele Anwender hegen den Wunsch, ihre 3D-Animationen auch ins Internet zu bringen. Da das Medium Internet allerdings andere Ansprüche an Dateigröße und -format stellt, müssen 3D-Anwender Kompromisse schließen – und sich das Flash-Format genauer ansehen.

GUT OPTIMIERTE UND KOMPRIMIERTE 3D-ANIMATIONEN im QuickTime- oder AVI-Format sind zwar in ihrer Qualität brillant, dennoch ist die Bandbreite des Internets ein arger Flaschenhals für solche animierten Webinhalte.

Als Standard für bewegte Internet-Seiten hat sich das vektororientierte Flash-Format fest etabliert. Die Animation beschränkt sich dabei auf zwei Dimensionen und besticht dank des Vektorformats durch äußerst kleine Dateigrößen und Skalierbarkeit.

Nun ist der logische nächste Schritt, das Flash-Format auch 3D-Applikationen als Ausgabeformat zugänglich zu machen. Dafür gibt es zwei wesentliche Vorgehensweisen.

Seit Version 8 gehört das FlashEx-Plug-in zum Lieferumfang von Cinema 4D. Über dieses Plug-in können Sie Ihre Objekte, Szenen und Animationen in das vektorbasierte Flash-Format ausgeben.

Eine andere Möglichkeit ist der Export der 3D-Objekte in eine spezialisierte 3D-Flash-Software wie Swift 3D oder E-Picture Pro. Dort animieren und optimieren Sie die Szene, um sie schließlich als Shockwave-Flash-Datei abzuspeichern.

Ich werde im Folgenden beide Vorgehensweisen vorstellen, damit Sie selbst entscheiden können, ob und welche Variante sich für Ihre Zwecke am besten eignet.

FlashEx

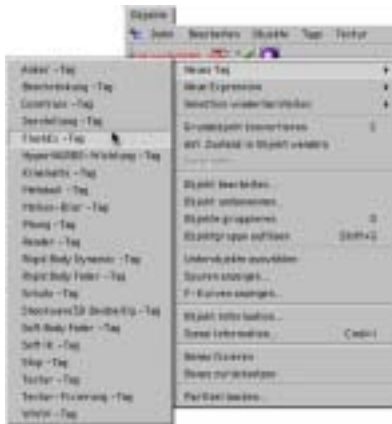
Endlich ist es so weit: Cinema 4D kann vektorbasierte Flash-Dateien und -Animationen rendern. Wer aber meint, er könne seine Werke per Knopfdruck automatisch in perfekte Flash-Kunstwerke verwandeln, sieht sich getäuscht. Der Export nach Flash bedarf einiger Vorarbeit und bringt prinzip- bzw. formatbedingt einige Einschränkungen mit sich.

Vor der Umwandlung sollten Sie Ihre Objekte bzw. Szene für den Flash-Export optimieren, andernfalls kann es nicht nur zu unerwarteten Ergebnissen oder langwierigen Renderzeiten, sondern sogar zum Renderabbruch kommen.

Zur Optimierung gehört, die Szene um die Objekte bzw. Materialien zu erleichtern, die FlashEx nicht übertragen kann, dazu gehören Szene-Objekte wie das Himmel-Objekt,



▲ **Abbildung 1**
FlashEx-Einstellungen Allgemein



◀ **Abbildung 5**
FlashEx-Tag



◀ **Abbildung 6**
FlashEx-Tag
Einstellungen



◀ **Abbildung 2**
FlashEx-Einstellungen
Hintergrund



◀ **Abbildung 3**
FlashEx-
Einstellungen
Linien

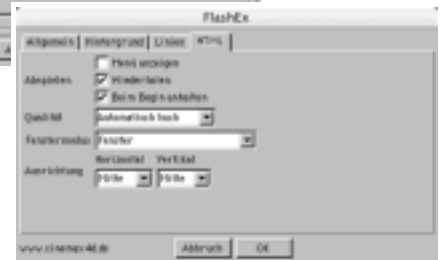


Abbildung 4 ▲
FlashEx-Einstellungen HTML

Materialien mit Bitmap-Texturen, Transparenzen und Relief-Kanälen, 2D- und 3D-Shader sowie andere Kameraeinstellungen als die Zentralperspektive.

Ist die Szene so weit vorbereitet, läuft das eigentliche Rendering über das FlashEx-Plug-in im Menü PLUG-INS ab. Auf der Allgemein-Seite des FlashEx-Dialogs (Abbildung 1) tragen Sie die Animationslänge, die Bilderrate und die vorgesehene Idealgröße der Flash-Datei ein. Im Feld Modus wählen Sie den gewünschten Rendermodus aus, eine Übersicht dazu sehen Sie auf der nächsten Seite.

Die Hintergrund-Seite (Abbildung 2) lässt Sie über die gewohnten Schieberegler einen Fond für das Flash-Rendering festlegen.

Auf die gleiche Weise bestimmen Sie auf der Linien-Seite des Dialogs die Farbe bzw. Transparenz der Umrisslinien für die Objekte. Sollen Linien gezeichnet werden, aktivieren Sie die Option UMRISSE und geben eine LINIENBREITE Ihrer Wahl an. Der KANTEN-WINKEL sagt aus, ab welchem Winkel aneinander liegende Polygone als Kanten gelten. Mit der Option FLÄCHEN ZUSAMMENFASSEN schalten Sie unschöne Polygonlinien bei NURBS- oder Boole-Objekten aus, die im Wireframe-Modus störend wirken würden.

Auf Wunsch kann FlashEx mit dem Export eine HTML-Datei erzeugen, in welche die entstandene SWF-Datei gleich eingebettet ist (Abbildung 1). Die Parameter für die Einbet-



▲ **Abbildung 7**
Wireframe



▲ **Abbildung 8**
Hiddenline



▲ **Abbildung 9**
Outline



▲ **Abbildung 10**
Solid



▲ **Abbildung 11**
Gouraud limitiert



▲ **Abbildung 12**
Comic flach limitiert



▲ **Abbildung 13**
Comic rund limitiert

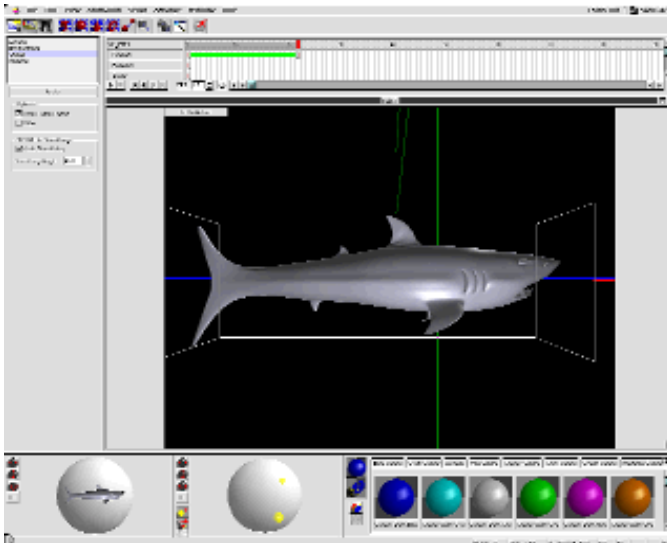
tung finden Sie auf der HTML-Seite des FlashEx-Dialogs (Abbildung 4).

Die bisher behandelten Optionen galten als globale Flash-Renderparameter. Natürlich können auch objektspezifische Renderkriterien verliehen werden. Dafür ist das FlashEx-Tag zuständig, das Sie dem jeweiligen Objekt im Objekt-Manager über DATEI · NEUES TAG (Abbildung 5) zuweisen. In dessen Einstellungsdialog (Abbildung 6) finden Sie objekteigene Linienparameter, in denen Sie Farbe, Transparenz, Stärke, Kanten-Winkel etc. für speziell dieses Objekt abstimmen.

Soweit die Einstellungsmöglichkeiten, die Ihnen das FlashEx-Plug-in für die Ausgabe anbietet. Kommen wir noch einmal auf die Rendermodi zurück, die Sie auf der Allgemein-Seite des FlashEx-Dialogs auswählen. Auf dieser Buchseite sehen Sie alle sieben verfügbaren Rendermodi aufgereiht (Abbildungen 7–13).

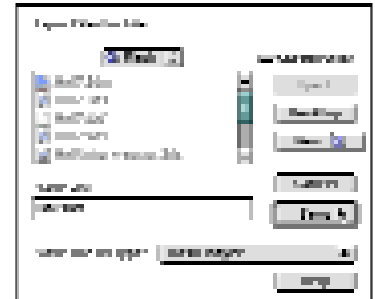
Im Wireframe-Modus (Abbildung 7) erscheinen die Objekte als Drahtgitter, wahlweise mit oder ohne Objektrückseiten. Der Hiddenline-Modus (Abbildung 8) unterscheidet sich lediglich durch die Undurchsichtigkeit der Objekte, der Outline-Modus (Abbildung 9) zeichnet ausschließlich die Objekturnrundungen nach. Eine zusätzliche Füllung des Objektes mit der Materialfarbe bewirkt der Solid-Modus (Abbildung 10). Diese vier Rendermodi gibt FlashEx komplett vektorbasiert und damit voll skalierbar und ressourcenschonend aus.

Die restlichen Rendermodi (Abbildungen 11–13) greifen dagegen auf Bitmap-Pixel zurück und empfehlen sich daher eher zur Simulation eines bestimmten Zeichentrick-Stils als zum Flash-Export in Hinsicht auf Dateigrößen- bzw. Bandbreitenschonung und Skalierbarkeit. Am besten, Sie sammeln einfach Ihre eigenen Erfahrungen mit dem FlashEx(port)-Plug-in.



◀ **Abbildung 14**
Cinema 4D-Objekt in
Swift 3D

Export der Animation
ins Flash-Format
Abbildung 15 ▼



Swift 3D/E-Picture Pro

Einen unkomplizierten Weg für die Verarbeitung von 3D-Daten in das Flash-Format versprechen spezielle 3D-Flash-Programme wie Swift 3D und E-Picture Pro.

Dazu speichern Sie Ihre in Cinema 4D erstellten Modelle in den Formaten 3D-Studio (.3ds) oder DXF (.dxf) ab und importieren sie in die Flash-Animatoren.

Dort haben Sie dann die Möglichkeit, Ihre Modelle mit rudimentären Animationswerkzeugen zu animieren. Auch hier erfolgt die Animation über eine Zeitleiste, wo Sie die Eigenschaften der importierten Objekte key-frame-basiert verändern und steuern. Wie bereits erwähnt sind die Animationsfunktionen relativ einfach gestrickt – für komplexe Animationen ist keines der Programme ausgelegt, erwarten Sie also nicht zu viel.

Zudem sollten auch die Modelle nicht allzu aufwändig sein – Polygon-Modelle lassen sich

über spezielle Optimierungssoftware wie Hexa-
super vorher überarbeiten.

Die Oberfläche (Abbildung 14) ist recht übersichtlich gestaltet, Texturen können Sie wie gewohnt per Drag and Drop vergeben und auch den Lichteinfall bestimmen. Das Arbeiten in beiden Programmen erfolgt weitgehend intuitiv. Die fertige Animation lässt sich anschließend ins Flash-Format (Abbildung 15) exportieren.

Insbesondere bei der Arbeitsgeschwindigkeit merkt man beiden Programmen an, dass ihnen eigentlich lieber wäre, Objekte der haus-eigenen Palette zu verarbeiten.

Swift 3D und E-Picture Pro sind gute Lösungen, um simple 3D-Modelle mit einfachen Animationen zu versehen und in das Flash-Format zu bringen. Allerdings dürfte als Zielgruppe wohl eher der 3D-interessierte 2D-Grafiker als der ambitionierte 3D-Designer, der Flash-Animationen erstellen will, angesehen werden.