
REFTOOLS!



Jegliche Programme und Dokumentationen sind Urheberrechtlich geschützt und nur für den „privaten Gebrauch“ bestimmt. Das Copyright liegt bei den jeweiligen Autoren. Das Vervielfältigen und Weiterreichen (CD, Diskette und/oder Internet), mit der Absicht der Massenverbreitung, ist ausdrücklich untersagt, solange nicht die Zustimmung des jeweiligen Autors vorliegt. Die Autoren übernehmen keine Verantwortung für Schäden und Folgeschäden, die durch eines der Programme verursacht worden sein könnten.

Kontakt: D. Kamp, Viehoferplatz 15, 45127 Essen

©1998 by REFTools!

Inhaltsverzeichnis

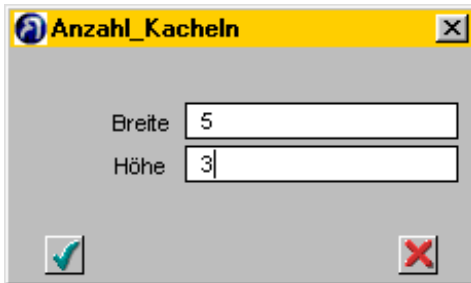
TEXTUR-KACHELER	4
VERSATZFREI-KACHELER	5
FLACKER	6
PLASMA-RING	8
VR-PLAYER	11
SUPER TORUS	14
2D-KURVEN	16
FUNC-FX	20
DUST	26
GALAXY	27
POINT-ANIMATION	28
POLYGON ZU KUGELOBJEKT	29
SET2ZERO	30
OBJ2PRT	31
TEXTUREMAGIC 3D 1	32
LANDSCAPE	40
PRTPARAM	42
REFTools!team	43

**Für Inhaltliche-, Grammatische- und Rechtschreibfehler ist der
jeweilige Autor verantwortlich.**

TEXTUR-KACHELER

Mit dieser Operation läßt sich eine Textur versatzfrei auf einem Objekt „kacheln“, d. h. das Texturbild wird passend verkleinert und mehrfach um bzw. auf das 3D-Objekt projiziert, so daß das Texturbild immer vollständig dargestellt wird. Wie oft das Bild verwendet werden soll, kann vom Anwender in einem Eingabefenster festgelegt werden.

Der Modus (Fläche / Kugel / Zylinder) ist beliebig.



Ablauf:

1. Wählen Sie mit der Bearbeiten-Funktion ein Textur-Objekt aus oder ziehen Sie das Bearbeiten-Drag&Drop auf ein Textur-Objekt.
2. Wählen Sie im Bearbeiten-Popup: Operation > Kacheln.
3. Im folgenden Eingabefenster geben Sie an, wie oft das Textur-Bild in der Breite und in der Höhe wiederholt werden soll. (Minimum bei den Werten ist 1!) Bestätigen Sie mit dem OK-Symbol.
4. Jetzt werden die Verkleinerungs-Faktoren berechnet und die Textur entsprechend angepaßt.

Das Skript ist so programmiert, daß es aus dem Bearbeiten>Operation-Popup für Textur-Objekte aufgerufen werden kann.

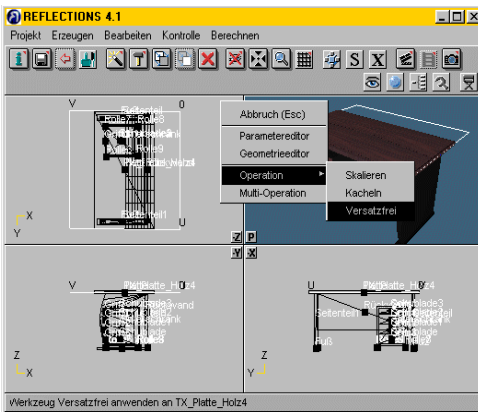
Damit das funktioniert, muß folgender Eintrag im Start-Skript (START.SKR im \REF\SKRIPTS-Verzeichnis) ergänzt werden:

```
ADD_SINGLE_TOOL_SKRIPT skripts\Texkach.tol 15 Kacheln  
Texkach.doc
```

VERSATZFREI-KACHELER

Mit dieser Operation läßt sich eine Textur versatzfrei auf einem Objekt „kacheln“, d. h. das Texturbild wird passend verkleinert und mehrfach um bzw. auf das 3D-Objekt projiziert, sodas das Bild immer vollständig dargestellt wird. Dabei wird die Bildgröße (möglichst) übernommen, die vom Anwender im Textur- Parameterfenster per DU- und DV-Regler annähernd eingestellt wurde.

Der Modus (Fläche / Kugel / Zylinder) ist beliebig.



Ablauf:

1. Wählen Sie mit der Bearbeiten-Funktion ein Textur-Objekt aus oder ziehen Sie das Bearbeiten-Drag&Drop auf ein Textur-Objekt.
2. Wählen Sie im Bearbeiten-Popup: Operation > Versatzfrei.
3. Jetzt werden die Verkleinerungsfaktoren berechnet und die Textur entsprechend angepaßt.

Das Skript ist so programmiert, daß es aus dem Bearbeiten>Operation-Popup für Textur-Objekte aufgerufen werden kann.

Damit das funktioniert, muß folgender Eintrag im Start-Skript (START.SKR im \REF\SKRIPTS-Verzeichnis) ergänzt werden:

```
ADD_SINGLE_TOOL_SKRIPT skripts\Txvers.tol 15 Versatzfrei
Txvers.doc
```

FLACKER

Mit dem „Flacker“-AnimFX kann man eine Lichtquelle „flackern“ lassen.

Der „Flacker“-AnimFX kann über den „Intensität“-Parameter animiert werden:

Intensität: Hier kann festgelegt werden, wie intensiv die Lichtquelle flackern soll. Wenn der Wert auf 0 steht, flackert die Lichtquelle überhaupt nicht; bei einem Wert von 1 wird die Lichtquelle unter Umständen komplett dunkel. Da die Lichtquelle durch Einsatz dieses AnimFX nie heller wird, sollte man beim Erzeugen der Lichtquelle die maximal gewünschte Helligkeit voreinstellen.

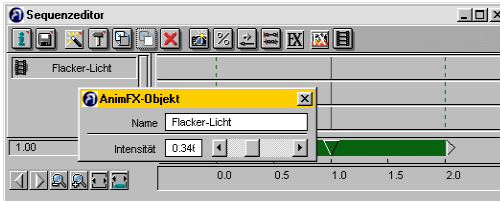
Der „Flacker“-Effekt wird wie üblich animiert, indem im Parametereditor ein entsprechender Wert vergeben und ein Snap gesetzt wird. Es können natürlich noch weitere Snaps mit anderen Intensitäts-Werten gesetzt werden.

Hinweis: Snaps müssen nicht unbedingt gesetzt werden. Soll die Lichtquelle kontinuierlich „flackern“, genügt es, den Parameter einmal einzustellen.

Anwendung:

Erzeugen Sie zunächst eine Lichtquelle und klicken Sie anschließend im Sequenzeditor auf die „Spezialeffekte“-Schaltfläche. Wählen Sie im folgenden Popup-Fenster den „Flacker“-Effekt aus. Wählen Sie nun den Lichtquellentyp aus, den Sie animieren möchten. Hierbei wird unterschieden nach „Punkt/Spot/Lokal/LokalSpot“ oder „Verteilter“ Lichtquelle. Befinden sich mehrere Lichtquellen in der Szene, werden Sie gegebenenfalls aufgefordert eine auszuwählen. Sollte sich keine Lichtquelle der ausgewählten Lichtquellenart in der Szene befinden, wird das AnimFX beendet ohne eine Sequenz zu erzeugen. Haben Sie ein Lichtquellen-Objekt ausgewählt, legen Sie nun fest, wie intensiv die Lichtquelle flackern soll, indem Sie einen entsprechenden Wert bei Intensität eintragen, z.B. 0.5 - d. h. die Helligkeit wird maximal bis auf die Hälfte der Ausgangshelligkeit vermindert.

Klicken Sie auf die „Vorschau-Animation“-Schaltfläche um einen Eindruck über die Auswirkung des Effekts zu erhalten.



Tips und Tricks:

Flackern in einem bestimmten Zeitraum:

Setzen Sie die Intensität auf 0 und erzeugen Sie einen Snap. Ändern Sie die Snap-Eigenschaft auf „Sprunghaft“, indem Sie das Bearbeiten Drag&Drop auf den Snap ziehen.

Erzeugen Sie nun einen Snap an der Stelle, ab welcher der „Flacker“-Effekt einsetzen soll. Legen Sie im Parametereditor die gewünschte Intensität, z.B. 0.7, fest. In einem weiteren Snap können Sie das Flackern wieder beenden, indem Sie die Intensität zurück auf 0 stellen. Die Lichtquelle wird nun während des festgelegten Zeitraums mit der eingestellten Intensität flackern.

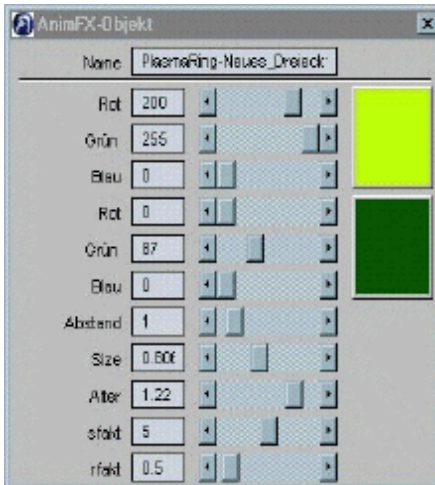
Offenes Feuer bzw. Flamme simulieren:

Um den Eindruck eines Feuers oder einer Flamme zu erzeugen empfiehlt es sich, zusätzlich zum „Flacker“-Effekt den „Rüttel“-Effekt auf die Lichtquelle anzuwenden. Dadurch erreicht man, dass sich neben der veränderten Helligkeit auch die geworfenen Schatten „bewegen“.

PLASMA-RING

Mit dem AnimFX „PlasmaRing“ können Sie eine brennende Plasma-Ring Explosion erzeugen. Die Erzeugung geschieht unter der Verwendung des Partikelsystems. Dabei werden am Ursprungspunkt eines beliebigen Geometrieobjektes „glühende“ Partikel erzeugt. Diese werden vom Ursprungspunkt ringförmig in X-Y Richtung fortbewegt und erzeugen hierbei eine Flammen (Plasma) Spur. Überschreiten die Partikel ihr Alter, werden sie gelöscht.

Beim AnimFX „PlasmaRing“ sind 11 animierbare Parameter vorhanden:



Rot, Grün, Blau:

Hiermit wird die Farbe des PlasmaRings eingestellt, was auch die Anfangsfarbe der Flammenpartikel ist.

Rot 2, Grün 2, Blau 2:

Hiermit wird die Endfarbe der Flammenpartikel eingestellt.

Abstand:

Hiermit wird der Abstand der Ring-Partikel zu einander eingestellt. Der Wert 1 bedeutet das sich die Partikel gerade noch berühren. Dieser Wert sollte

nicht zu klein gewählt werden, da sonst sehr viele Partikel erzeugt werden. Ein Wert von 0 erzeugt keine Partikel.

Size:

Hiermit wird die Größe der Partikel festgelegt.

(Als Richtwert: Size 1 wenn Objekt ca. 100 Einheiten groß ist.)

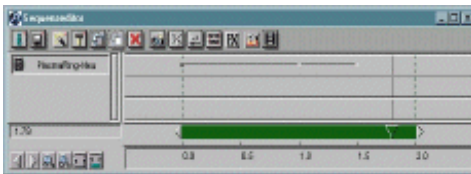
Alter:

Hiermit wird das Alter der Plasma-Spur eingestellt. Dabei wird das Alter zufällig an die Partikel verteilt, damit das Verhalten nicht zu gleichförmig ist. Mit dem Alter hat man auch Einfluß auf die Länge der Plasmaspur. Überschreiten die Partikel ihr Alter, werden sie gelöscht.

sfakt / rfakt:

Mit zunehmendem Alter wird auch die Stärke und der Radius der Flammen Partikel verändert. Diese beiden Parameter beeinflussen wie stark dies geschieht. Mann kann somit eine rauchähnliche Wolke oder auch eine Plasmaspur aus „Festen“- Partikel erzeugen.

Animiert wird der AnimFX „PlasmaRing“ wie jedes andere Objekt auch. Im Parametereditor des AnimFX setzen Sie die Parameter auf bestimmte Werte und erzeugen einen Snap. Ändern Sie die Parameter erneut und setzen einen neuen Snap. Diese Prozedur können Sie entsprechend fortsetzen.



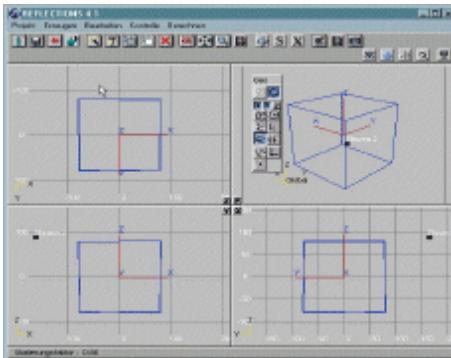
Hinweis :

Snap müssen nicht unbedingt gesetzt werden.

Besonderheiten:

Jeder Glühpartikel erzeugt bei jedem Bild einige Plasmateilchen an seiner Position, bevor es weiterbewegt wird. Dadurch können mit der Zeit sehr viele Partikel erzeugt werden, die schon gewisse Ressourcen (Speicher, Rechenzeit) benötigen.

Die Anzahl der Rauchpartikel kann begrenzt werden, indem das Alter niedriger gesetzt wird (Somit werden die Rauchpartikel schneller gelöscht) oder der Abstand vergrößert wird. Wird der Parameter „Abstand“ auf 0 gesetzt, werden keine Partikel erzeugt.



Wird das Geometrieobjekt gelöscht, welches dem AnimFX zugewiesen ist, so wird auch der AnimFX gelöscht. Die Partikel werden vom Ursprungspunkt eines Geometrieobjektes in XY-Richtung ausgestrahlt. Es empfiehlt sich, hierzu ein Dreiecksobjekt „Frei“ zu verwenden.

Die Größe und Ausdehnungsgeschwindigkeit der Partikel wird durch die Größe des Geometrieobjektes bestimmt. Wenn Sie das Geometrieobjekt im Geometrieditor bewegen oder skalieren, sehen Sie eine Box. Die Boxgröße entspricht der Entfernung, welche die schnellsten Glühpartikel innerhalb einer Sekunde erreichen.

Anwendung:

Erzeugen Sie ein neues Dreiecksobjekt „Frei“.

- Vergrößern sie das Objekt auf ca. 100 Einheiten !!! (wichtig sonst sind die Partikel zu groß und man sieht nur ROT)

- Wechseln Sie in den Sequenzeditor. Hier klicken Sie mit der linken Maustaste auf den Funktionsschalter „Spezialeffekte“. In dem sich öffnenden Auswahlrequester wählen Sie „PlasmaRing“ aus. Falls sich mehrere Objekte in der Szene befinden, wählen Sie in dem nun folgenden Auswahlrequester „Neues Dreieck“ aus.

- Im Anschluß öffnet sich der Parametereditor des AnimFX. Hier geben Sie bitte bei „Rot“, „Grün“ und „Blau“ einen Farbwert an.

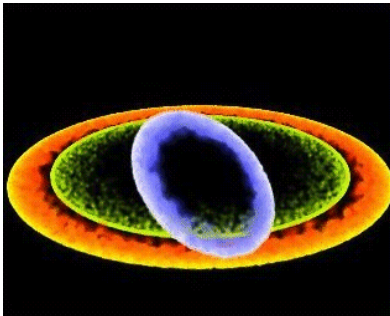
- Für „Abstand „ geben Sie den Wert 1,5 und

- Für „Size“ den Wert 1 und

- Für „Alter“ einen Wert von 0,5 ein.

- sfakt und rfakt sollten den Wert 1 bekommen.

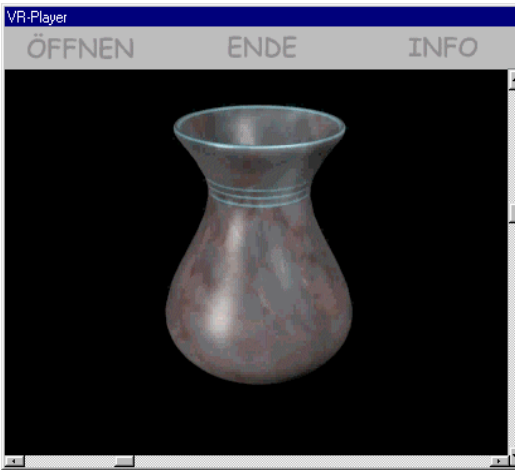
- Klicken Sie nun mit der linken Maustaste auf den Funktionsschalter „Spezialeffekt rendern“. Nach kurzer Vorberechnungsphase startet BEAMS mit der eigentlichen Bildberechnung. Beachten Sie aber, daß für eine Vorschauberechnung die AnimFX-Sequenz selektiert sein muß.



VR-PLAYER

Der VR-Player (PC) ist ein Tool zum betrachten von VR-Dateien im AVI-Format. Damit ist es z.B. möglich ein Objekt in Echtzeit zu drehen, und das um beide Achsen.

Eine VR-Datei besteht aus vielen berechneten Bildern die in einer bestimmten Reihenfolge zusammengesetzt sind.



Die Oberfläche ist sehr einfach gehalten.

Erstellen einer VR-Datei:

Laden wir dazu die mitgelieferte Beispielszene „VR.R4“.

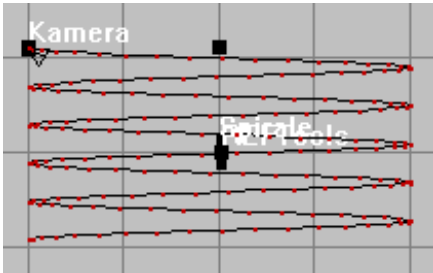
Das Objekt „REFTools“ befindet sich genau am Ursprung, das macht uns die Arbeit etwas leichter.

Erzeugen Sie jetzt ein 3D-Polygon „SPIRALE“ mit folgenden Parametern.

Machen wir uns jetzt schnell ein paar Gedanken über die wichtigen Parameter. Dazu gehört die Kreisanzahl, Spiralhöhe und der Spiralaradius. Je mehr die Kreisanzahl steigt, um so mehr Ebenen können zum Betrachten verwendet werden. Wir wählen für unser Beispiel 5 Kreise. An der Stelle für Kreispunkte tragen wir immer 36 ein. Die Spiralhöhe richtet sich nach dem gewünschten Höhenunterschied der Kameraanfangsposition und der Endposition, für uns jetzt -10. Minus deshalb, damit die Kamera in die korrekte Richtung bewegt wird. Der Spiralaradius ist abhängig von dem Objekt oder der Szene um die die Kamera schweben soll. Der Abstand 10 ist für unsere Szene ausreichend.

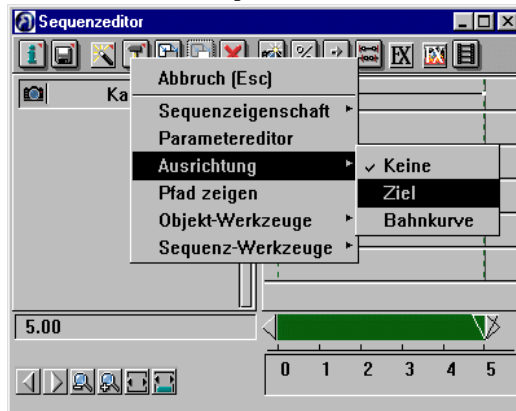
REFTOOLS!

Nach der Bestätigung zum Erstellen der Spirale verschieben wir diese in der Höhe, bis die Spirale genau in der Mitte der Szene platziert ist. Bewegen wir nun die Kamera zum Anfangspunkt. Der Anfangspunkt ist immer oben der erste Punkt der Spirale.

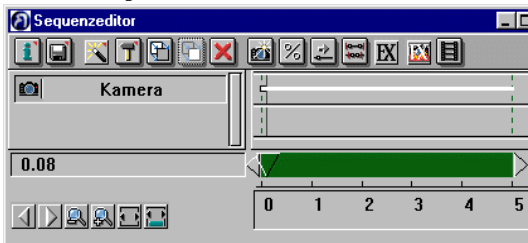


Wenn Sie alles richtig gemacht haben, sollte das Ergebnis wie in Beispiel „VR1.R4“ aussehen. Öffnen wir jetzt den Sequenzeditor und erzeugen eine Sequenz für die Kamera. Dazu stellen wir die Länge auf 5 (Kreisanzahl) ein. Nun noch die Snaps auf 0 und 5 setzen.

Über das „Bearbeiten“-Icon gehen wir zu „Ausrichtung-Ziel“. Bei dem jetzt geöffneten Fenster wählen wir das Objekt „REFTools“ (Objekt im Zentrum). Nun können wir den Sequenzeditor schließen. Weiter geht es im Operations-Manager. Hier klicken wir auf den



Button „+Spirale“ und schalten den Abschnitt „3D-Linie zur Sequenz“.



Die fertige Sequenz sieht dann so aus (VR2.R4)

Zum erstellen der fertigen VR-Datei fehlt uns nur noch die Berechnung.

The screenshot shows the 'Animationsjob' dialog box. It has a title bar with a blue gradient and a close button. Below the title bar is a toolbar with a small icon. The main area contains several input fields and buttons. At the top, there are two text boxes: 'Jobpfad' with the value 'd:\ref41\szenen\vr2.job' and 'Szenenpfad' with the value 'd:\ref41\szenen\vr2.r4'. Below these are two buttons: 'Beams' and 'Seq-Intervall'. Further down are two text boxes for 'Startzeit' (0.000381) and 'Ende' (5.001399). Below these are two text boxes for 'Bildanzahl' (180) and 'Bildnr: 0'. In the center, there is a label 'Anim-Typ' followed by a dropdown menu showing 'AVI-24bit'. Below this is a text box for 'Anim-Datei' with the value 'vr-beispiel.avi'. At the bottom, there is a section labeled 'Ruht' with three buttons: 'Neu', 'Sichern', and 'Start'. The 'Start' button is highlighted with a darker background.

Dazu öffnen wir das Animationsjob-Fenster. Stellen die gewünschten Beams-Parameter ein (nicht größer 400x300 Punkte), auf „Seq-Intervall“ klicken und die Bildanzahl eintragen. diese berechnet sich aus 36 und der Kreisanzahl, also $36 \cdot 5$. Das sind 180 Bilder.

Als Anim-Typ schalten wir auf AVI. Nun schreiben wir noch den Animationsnamen ein.

Zum Beginnen der Berechnung klicken wir auf den Button „Start“. Nach der Berechnung steht uns eine VR-Datei des berechneten Objektes zur Verfügung.

SUPER TORUS

Mit Hilfe dieses Plugins lässt sich ein Torus erstellen, der sich vom Reflections-Standardtorus durch eine flexiblere Formgebung unterscheidet. Man kann der äußeren Ringform sowohl eine quadratische als auch eine sternförmige Ausprägung geben. Neben der Gestalt der Seiten lässt sich auch die Auflösung des Toroids einstellen.

Hinweise:

- Bei den Dreiecken des erzeugten Objektes wird standardmäßig das Rund-Flaggesetzt.

- An den Kanten des Objektes wird je nach Form eine höhere Anzahl von Dreiecken erzeugt als an flachen Stellen.

- Ein „normaler“ Torus ergibt sich, wenn man Ringform und Rohrform jeweils auf 1 setzt.

- Das Objekt wird nicht automatisch zentriert (wie bei den in Reflections eingebauten Objekten). Dies ist allerdings kein Fehler in SuperTorus, sondern war zum Zeitpunkt der Erstellung des Plugins nicht möglich (jetzt, wo Sie dies lesen, allerdings schon). Ein Update hierzu (wie zu allen Plugins, bei denen dies notwendig ist) wird über die Reftools-Homepage erhältlich sein. In der Zwischenzeit müssen Sie das Objekt je nach Situation mit Hilfe der Maus oder per Tastatur „von Hand“ zentrieren.

Installation (falls sie von Hand installieren):

Bitte kopieren Sie die Datei SuperTorus.dll nach

<refverzeichnis>\plugin\startup

und erstellen Sie das Verzeichnis

<refverzeichnis>\plugin\data\supertorus

Falls dieses Verzeichnis nicht existiert, kann das Plugin die Konfiguration nicht speichern und beim Schließen von Reflections erscheint die Fehlermeldung "Kann Konfigurationsdatei nicht öffnen".

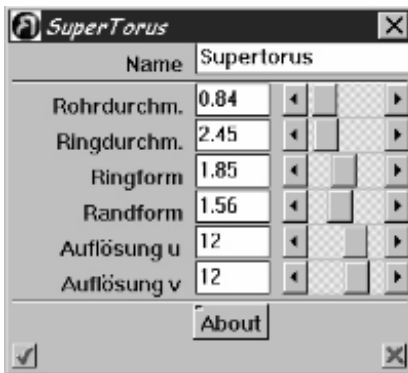
Wo finden Sie das Plugin?

Unter "Erzeugen Button" -> Plugins -> SuperTorus -> SuperTorus

Der zusätzliche Untermenüpunkt lässt sich nicht vermeiden.

Die Parameter:

Name: Objektname
 Rohrdurchm.: Durchmesser des Rohres
 Ringdurchm.: Durchmesser des gesamten Toroids
 Ringform: Hiermit läßt sich die Form des Ringes einstellen:
 rechteckig über rund bis sternförmig
 Auflösung u: Ringsegmente (1..17, spezieller Wert, nicht die Anzahl der Segmente!)
 Auflösung v: Rohrsegmente (1..17, "''''")



2D-KURVEN

Mit Hilfe dieses Skriptes können in Reflections verschiedene zweidimensionale Kurven als Polygone erzeugt werden:

- Lissajous-Figur
- Zykloid
- Epizykloid
- Hypozykloid
- Epitrochoid
- Hypotrochoid
- Nephroid
- Astroid
- Trikuspid



Mit Hilfe dieser Kurven können z.B. Schläuche oder auch andere dreidimensionale Objekte erstellt werden.

Es liegen 4 Dateien bei:

- 2dkurven.txt (diese Beschreibung)
- 2d-kurven.men (Hauptskript)
- getcps.skr (Unterskript zum Einlesen der Kurvenparameter)
- f_draw.skr (Unterskript zum Zeichnen der Kurven)

Grundsätzliches:

Nach dem Start des Skriptes werden Sie gefragt, welche KURVE erzeugt werden soll.

Anschließend müssen Sie einen Wert für die OBERE BERECHNUNGSGRENZE (og) eingeben. Dieser Wert (>0) bestimmt, wie weit die Kurve gezeichnet werden soll. Wenn der Wert zu klein ist, bleibt ist die Kurve offen; wenn er zu groß ist, wird die Kurve (oder ein Teil davon) mehrfach gezeichnet. Wenn Sie sich nicht ganz sicher sind, wählen Sie im Punkteditor "Polygon schließen". Dadurch sieht man es fast immer, weil damit der Anfang direkt mit dem Ende verbunden wird.

Für diese Variable kann kein genauer Wert angegeben werden, da er bei den meisten Kurven von den eigentlichen Kurvenparametern abhängt. Beachten Sie, dass eine höhere Grenze eine höhere Rechenzeit zur Folge hat.

Darum sollten Sie bei der letzten allgemeinen Einstellung, der AUFLÖSUNG der Kurve, mit der Einstellung "Grob" beginnen, da die Rechenzeit dann am kürzesten ist. Über die Auflösung bestimmen Sie die Schrittweite, mit der die Kurve gezeichnet werden soll, also letztendlich die Anzahl der Punkte.

Es ist möglich, dass die Kurve auch bei richtiger oberer Berechnungsgrenze nicht ganz geschlossen ist. Dann (und eigentlich auch sonst, wenn eine geschlossene Kurve erzeugt wurde) sollte man das Polygon im Punkteditor mit Hilfe der entsprechenden Funktion schließen.

Es folgen die Infos zu den einzelnen Kurven:

Lissajous-Figur:

Eine Lissajous-Figur entsteht durch die Überlagerung harmonischer Schwingungen bei senkrecht aufeinander stehenden Schwingungsrichtungen. Diese Kurven werden in der Astronomie, Physik und anderen Wissenschaften verwendet. Untersucht wurden sie zuerst 1815 von Nathaniel Bowditch und später von Jules-Antoine Lissajous (1857).

Beispiele für Parameter:

- "8": $og=2, a=2, b=0$
- Oval: $og=2, a=1, b=2$
- Kreis: $og=2, a=1, b=PI/2=1.570796$
- $og=2, a=2, b=1$
- $og=10, a=1.6, b=1$

Zykloid:

Der Weg eines Punktes von einem Kreises der auf einer Geraden ohne zu gleiten abrollt.

Beispiel für Parameter:

- $og=4, a=2, b=2$

Trikuspoid:

Diese Kurve bildet eine Art Dreieck mit nach innen abgerundeten Seiten. OG ist hierbei immer 2 für eine geschlossene Kurve. Mit dem Parameter a bestimmen Sie die Größe.

Man kann hier auch eine Seite im Punkteditor löschen und den Rest zur Konstruktion von Objekten verwenden.

Epizykloid:

Der Weg eines Punktes des Umfangs eines Kreises, der ohne zu gleiten auf

der Aussenseite eines festen Kreises rollt.

a =Radius des festen Kreises

b =Radius des rollenden Kreises

Beispiel für Parameter:

- $og=10$, $a=8$, $b=5$

Hypozykloid:

Der Weg eines Punktes des Umfangs eines Kreises, der ohne zu gleiten auf der Innenseite eines festen Kreises rollt.

Beispiel für Parameter:

- $og=6$, $a=5$, $b=3$

Epitrochoid:

Beschreibung wie bei Epizykloid, aber der erzeugende Punkt liegt innerhalb oder außerhalb des rollenden Kreises im Abstand c vom Mittelpunkt des rollenden Kreises.

Beispiel für Parameter:

- $og=6$, $a=5$, $b=3$, $c=5$

Hypotrochoid:

Beschreibung wie bei Hypozykloid, aber der erzeugende Punkt liegt innerhalb oder außerhalb des rollenden Kreises im Abstand c vom Mittelpunkt des rollenden Kreises.

Beispiel für Parameter:

- $og=14$, $a=5$, $b=7$, $c=2.2$

Nephroid:

Ein Epizykloid mit folgenden Eigenschaften:

- rollender Kreis hat Radius a

- fester Kreis hat Radius $2*a$

Beispiel für Parameter:

- $og=2$, a =Größe

Astroid:

Aehnlich wie der Trikusoid, aber als Viereck.

Beispiel fuer Parameter:

- $og=2$, $a=Groesse$

FUNC-FX

Es liegen 2 Beispielszenen bei:

-swurf.r4: Schräger Wurf

-ffall.r4: Freier Fall

Dieses AnimFX-Skript bietet eine einfache(?) :) Möglichkeit, Position, Größe und Orientierung von Dreiecksobjekten, oder die Parameter eines Materials durch mathematische Funktionen zu steuern. Weil es eigentlich nicht möglich ist, Funktionen einigermaßen komfortabel in Skripts einzulesen (oder gar auszuwählen, was man eingeben will), wird dabei ein zugegebenermaßen etwas merkwürdiger Weg gegangen. Die Funktionen werden nämlich als Namen der Körper eines Objektes eingegeben und gespeichert, das vom Skript zu der entsprechenden Sequenz erzeugt wird.

Das hat allerdings auch einige kleine Nachteile:

-Wenn die Sequenz gelöscht wird, bleibt das "Speicher-Objekt" bestehen, muß also von Hand gelöscht werden. Leider geht es nicht anders. Wenn man es aber versehentlich nicht löscht und bereits eine neue FuncFX-Sequenz erzeugt hat, wird trotzdem ein neues Speicherobjekt erzeugt und richtig verwendet. Bitte achten Sie aber darauf, das Sie dann das RICHTIGE Speicherobjekt löschen, nicht das, welches für die neue Sequenz erstellt wurde. Denn dann wird die neue Sequenz gleich mitgelöscht. Wenn Sie sich nicht ganz sicher sind, speichern Sie die Datei einfach vor dem Löschen des Objektes. Die Namen der Speicherobjekte beginnen immer mit "FFFUNC-".

-Man muß aufpassen, daß man die Körpernamen im Speicherobjekt richtig benennt. Aber dies ist eigentlich nicht weiter schwierig, siehe weiter unten. Sie dürfen auch keinen der Einträge löschen.

-Die Länge der Formel ist begrenzt (Windows 60 Zeichen, AMIGA 100 Zeichen) und es ist bei einer längeren Formel nicht alles auf einmal zu sehen.

Tip: Schreiben und ändern Sie die Formeln in einem Texteditor und kopieren Sie sie dann per Copy&Paste in die Namensfelder.

Nun erstmal ein Beispiel:

-Löschen Sie die Szene, falls eine besteht

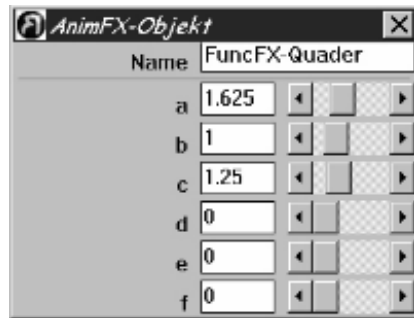
-Erzeugen Sie einen Quader

-Öffnen Sie das Sequenzfenster (über das Menü oder einen Rechtsklick auf das Sequenz-Icon im Hauptfenster)

- Klicken Sie auf das AnimFX-Icon und wählen Sie "FuncFX"
- Sie werden nun gefragt, ob Sie ein GeoObjekt oder ein Material animieren wollen. Wählen Sie "GeoObjekt" und anschließend "Quader"
- Öffnen Sie den Körpereditor zu "FFFUNC-Quader"

Sie sehen nun 10 Einträge:

MX cmx
 MY cmy
 MZ cmz
 RX crx
 RY cry
 RZ crz
 SX csx
 SY csy
 SZ csz
 BO -



Am Zeilenanfang steht jeweils ein Marker, welcher die zu steuernde Größe angibt. MX bedeutet z.B. "Position auf X-Achse". Diese Marker dürfen Sie nicht verändern. Auf sie folgt jeweils ein Leerzeichen und eine Funktion.

-Ändern Sie den Namen des MX-Körpers von "MX cmx" auf "MX 5*t" (ohne die Anführungszeichen). Dabei setzen Sie bitte kein Gleichheitszeichen, sondern geben nach dem "MX" und einem Leerzeichen die Formel ein. Alle anderen Einträge bleiben unverändert.

-Wenn Sie nun die Vorschauanimation ("aktueller Plotmodus") starten (z.B. mit 20 Bildern), sollten Sie sehen, wie sich das Objekt bewegt. Eventuell müssen Sie noch den Blickwinkel und/oder die Entfernung der Kamera zum Objekt ändern, um es optimal zu sehen.

"MX 5*t" bedeutet, dass die Position des Objektes auf der X-Achse dem Ausdruck "1.5*t" entsprechen soll. t ist dabei die Zeit des aktuellen Bildes. Bei einer Animation von 1 Sekunde Länge und 10 Bildern würde sich das Objekt auf der X-Achse in folgenden Schritten bewegen: 0, 0.5, 1, 1.5, 2, 2.5 usw. Der maximale X-Wert wäre $5 \cdot 1 = 5$.

Nun bewegt Sich das Objekt vielleicht zu schnell oder zu langsam, und Sie müßten für jede Änderung der Konstante (hier 5) die Formel im Körpereditor ändern.

Es gibt aber eine andere Möglichkeit:

Ändern Sie den Eintrag für MX ab zu "MX a*t" (also die 5 durch ein a ersetzen). Öffnen Sie dann im Sequenzeditor den Parametereditor zu FuncFX-Quader und stellen Sie für a einen Wert ein. Nun können Sie bequem die Konstante ändern, ohne jedesmal in den Körpereditor gehen zu müssen. Die Konstante "a" erhält bei der Animation den Wert, den Sie im Parametereditor zu FuncFX-Quader bei der Einstellung "a" eingegeben haben. Sie können beliebige Werte eingeben, die nicht durch den Schieber zu erreichen sind.

Um noch ein Detail auszuführen, bringen wir das Beispiel zu Ende:

- Schieben Sie den Quader mit Hilfe des Geoeditors auf eine andere Position (nur die Position auf dieser einen Achse verändern, also waagrecht im linken oberen Fenster bei der Standard-Ansicht)

- Starten Sie wieder die Animationsvorschau. Leider ist die Startposition des Quaders bei der Animation wieder bei 0, obwohl Sie ihn vorher verschoben haben. In einer richtigen Szene ist das natürlich recht ungünstig. Der Grund ist auch klar: Die Funktion berücksichtigt ja nicht die Position, auf den Sie den Quader vorher geschoben haben, sondern berechnet wie vorher "a*t".

- Öffnen Sie noch einmal den Körpereditor zu FFFUNC-Quader und ändern Sie "MX a*t" zu "MX a*t+cmx" ab.

- Starten Sie wieder die Animationsvorschau. Und siehe da: Das Objekt startet an der richtigen Stelle der X-Achse. Die c-Konstanten beinhalten nämlich immer den entsprechenden Startzustand des Objektes. Für jede zu steuernde Größe gibt es eine passende c-Konstante. Die "cm*", "cr*" und "cs*" -Konstanten müssen Sie jeweils zu Ihrem Ausdruck addieren (wie oben gezeigt),

Auch für alle nicht zu beeinflussenden Größen geben Sie bitte die entsprechende Konstante ein, damit sich dort nichts verändert. Nach der Erzeugung der Sequenz stehen allerdings hinter allen Markern schon die entsprechenden c-Konstanten.

Der letzte Eintrag ist "BO".

Dieser Marker beschreibt das Bezugsobjekt, dessen Namen Sie dort eingeben können, z.B. "BO Quader2". Wenn das Objekt nicht existiert, wird eine entsprechende Fehlermeldung ausgegeben. Wenn Sie kein Bezugsobjekt wollen, tragen Sie bitte wieder "BO -" ein, also BO, ein Leerzeichen und ein Minuszeichen.

Bleibt noch zu klären, was ein Bezugsobjekt ist:

Dieses Objekt liefert ihnen zusätzliche Variablen, die Sie in Ihren Formeln verwenden können:

Aktuelle Position des Bezugsobjektes:	omx, omy, omz
Aktuelle Orientierung des Bezugsobjektes:	orx, ory, orz
Aktuelle Größe des Bezugsobjektes:	osx, osy, osz

Wenn ihr FuncFX-Objekt ein Geoobjekt ist, kommen noch folgende Variablen dazu:

Aktueller Abstand zum Bezugsobjekt:	oeg
Aktueller X-Abstand zum Bezugsobjekt:	oex
Aktueller Y-Abstand zum Bezugsobjekt:	oey
Aktueller Z-Abstand zum Bezugsobjekt:	oez

(Das "e" soll "Entfernung" heißen)

Zum Schluß nun noch alle anderen Variablen, die Sie in Ihre Formeln einsetzen können:

Allgemein verwendbare Parameter:

pi (3.141...)

g (-9.81...)

bzg (Bogenmass zu Grad), Argument im Bogenmass damit multiplizieren, um Grad zu erhalten.

Objektvariablen (als Konstanten gedacht, kann man theoretisch aber auch über Snaps animieren)

a, b, c, d, e, f

Zeit t

Zufallswert zu (0 bis 1)

REFTOOLS!

GeoObjekt-Animation:

Aktuelle eigene Position:	amx, amy, amz
Aktuelle eigene Orientierung:	arx, ary, arz
Aktuelle eigene Größe:	asx, asy, asz
Eigene Ausgangsposition:	cmx, cmy, cmz
Eigene Ausgangsorientierung:	crx, cry, crz
Eigene Ausgangsgröße:	csx, csy, csz

Material-Animation:

Aktueller eigener Rotanteil:	aro
Aktueller eigener Grünanteil:	agr
Aktueller eigener Blauanteil:	abr
Akt. Diffusion:	adi
Akt. Spiegelung:	asp
Akt. Transparenz:	atr
Akt. Eigenleuchten:	aei
Akt. Brechungsindex:	abr
Ausgangs-Rotanteil:	cro
Ausgangs-Grünanteil:	cgr
Ausgangs-Blauanteil:	cbl
Ausgangs-Diffusion:	cdi
Ausgangs-Spiegelung:	csp
Ausgangs-Transparenz:	ctr
Ausgangs-Eigenleuchten:	cei
Ausgangs-Brechungsindex:	cbr

Einige Beispielfunktionen:

-Freier Fall entlang der Z-Achse:

$$MZ -g*(t^2)/2 \text{ oder } MZ -a*(t^2)/2$$

a müssen Sie dann mit dem Parametereditor im Sequenzeditor einstellen, wie im Beispiel des Quaders beschrieben.

-Senkrechter Wurf:

$$MZ a*t-(g*(t^2)/2)$$

-Waagerechter Wurf:

$$MX a*t$$

$$MZ -g*(t^2)/2$$

-Schräger Wurf:

$$MX \ a*t*\cos(b)$$

$$MZ \ a*t*\sin(b)-(g*(t^2)/2)$$

a = Geschwindigkeit

b = Abwurfwinkel

Hinweis: Sowohl das Skript als auch diese Beschreibung sind eine ,an einem Nachmittag entstandene, vorläufige Version.

Darum ist es durchaus möglich, dass sich Fehler eingeschlichen haben. Die Beschreibung ist auch nicht vollständig, da nicht auf Material-animationen eingegangen wurde, aber man kann es sich mit Hilfe der gegebenen Infos sicher zusammenreimen. Es fehlen auch Beispiele mit Bezugsobjekten wie z.B.

"Mitdrehen von Rädern", "Einfache Anziehung" und "Gegenseitige Anziehung". In der nächsten Version wird es vermutlich auch noch die eine oder andere Erweiterung geben.

Nachtrag:

Es wird noch nicht geprüft, ob Sie ein gültiges GeoObjekt wählen. Bitte verwenden Sie nur Materialien oder Dreiecksobjekte (z.B. keine Kamera und keine mathematische Kugel).

Bitte wählen Sie auch kein FFFUNC- Objekt, das hätte wenig Sinn. Außerdem sollten Sie darauf achten, daß in Ihrer Formel nichts durch 0 geteilt wird.

DUST

Diese Funktion erzeugt ein Staubfraktal aus echten Kugeln. Die erzeugten Kugeln sind in einem einzigen Objekt enthalten. Trotzdem können die enthaltenen Kugeln des Objektes, im Nachhinein, über den Ursprungspunkt einzeln bearbeitet werden (Materialvergabe, Größenänderung usw.).
Ablauf:

1. Der Name des zu erstellenden Dust-Objekts wird abgefragt.
2. Die fraktale Dimension in X-Richtung wird abgefragt.
3. Die fraktale Dimension in Y-Richtung wird abgefragt.
4. Die größte Expansion in Z-Richtung oder 0.
5. Der Berechnungswinkel in Grad wird abgefragt.
6. Die Anzahl von Staubpartikeln wird abgefragt.
7. Der Radius der Kugeln wird abgefragt.

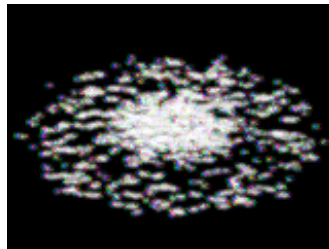
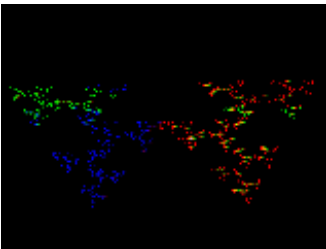
Algorithmus :

Zuerst einmal meinen Dank an Herrn Hans Lauwerier der zwei hervorragende Fraktalbücher geschrieben hat, auch wenn er dabei nur Intel-User unterstützt.

Die Zufallszahlen für das Staubfraktal entstehen mit Hilfe der Monte-Carlo-Methode, wobei zwei Drehungen berechnet werden. Darauf frei aufbauend, habe ich diese Funktion kreiert.

Die Fraktalen Dimensionen sind reelle Werte von 0 bis 1! Kleine Werte ergeben eine kleine Streuung, gleiche Werte ergeben ein symmetrisches Bild. Mit dem Berechnungswinkel werden die Startwerte sowie die Werte für die folgenden Iterationen berechnet. Er bestimmt die Grundform des Fraktals.

Bei gewissen Einstellungen wird eine recht hohe Anzahl von Staubpartikeln benötigt, um gute Resultate zu erhalten.



©REFTools!team/Stephan Biemann

GALAXY

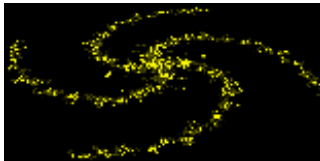
Diese Funktion erzeugt eine Galaxie aus echten Kugeln. Dies mit Hilfe eines simplen Algorithmus welcher von Nikola Smolenski stammt. Siehe auch das Programm Galaxy von Nikola Smolenski. Die erzeugten Kugeln sind in einem einzigen Objekt enthalten.

Trotzdem können die enthaltenen Kugeln des Objektes, im Nachhinein, über den Ursprungspunkt einzeln bearbeitet werden (Materialvergabe, Größenänderung usw.).

Im Moment werden 11 gleiche Materialien erstellt welche zufälligerweise den Sternen übergeben werden. Die aber noch manuell zu verbessern sind, in Farbe und Oberflächenverhalten.

Ablauf:

1. Der Name der Galaxie wird abgefragt.
2. Die Anzahl der Arme welche sie haben soll.
3. Den anfänglichen Durchmesser der Galaxie.
4. Den Ausbreitungswinkel der Arme.
5. Die Ausdehnung der Arme.
6. Die Dichte der Sterne.
7. Der minimale Radius der Sterne.
8. Der größte Unterschied zwischen minimalem und maximalem Radius der Sterne.
9. Die allgemeine Berechnungspräzision. Von 1 bis zum Ausbreitungswinkel.



POINT-ANIMATION

Ergebnis:

Ein Objekt (C), daß entsprechend einem Faktor (0%-100%) ein morphänliches zwischen Objekt A und Objekt B ist.

Wie man vorgeht:

Sie nehmen ein Objekt A und kopieren es zweimal (Objekt C und D) dann verformen Sie Objekt B (z.B. Punktverzerren).

Das Skript fragt Sie dann nach A,B und C ab; außerdem will es einen Wert zwischen 1 und 100 (Prozent) haben, den Sie bitte animieren.

Das Skript lässt dann die Punkte im Objekt D entsprechend wandern...

Bekannte Fehler:

Damit Sie im Wireframe Modus etwas sehen, muß das Skrip andauernd zwi-schen der Perspektiv und der 4 Fach -Ansicht wechseln. Das kostet Zeit, Nerven und flackert furchtbar. Wenn jemand eine Lösung kennt, dann her damit (Snag@gmx.de)

Zu der Demo:

Der Kopf ist ein Modell (Debbie) aus dem 3dcafe. Ich habe ihn mit der Platte verknüpft (A+B), das Ergebnis kopiert und mit dem Set2Zero-Skript auf Null gesetzt. Das dritte (sichtbare) Objekt wird dann das Ergebnis.

Tip:

Rendern Sie einfach einmal eine kleine 25 Bilder Anim und spätestens dann haben Sie gesehen warum und wie ich was gemacht habe.

P.S: die Anim kann nicht gut aussehen, da sich die „Platte“ nicht mitverzieht. (Aber für eine Demo reichts ja wohl)

POLYGON ZU KUGELOBJEKT

Was macht das Ding?

Ganz einfach: es setzt auf jeden Punkt von einem 2d- oder 3d- Polygon eine Kugel. (also ein richtiges „billig Skript“- aber ich hab’s halt gerade für was gebraucht und wollte es nun nicht der Öffentlichkeit vorenthalten)

Was will es wissen?

-Kugelradius: völlig egal, da Sie ihn ja jederzeit in Ref ohne Probleme ändern kannst.

-Kugelobjektnamen: Geben Sie den gewünschten Namen ein.

Zur Installation des Skripts:

Oh, das ist etwas komplizierter ;)

Sie müssen es in das Skripts Verzeichnis kopieren und dann im Start.SKR folgende Zeilen hinzufügen:

```
ADD_SINGLE_TOOL_SKRIPT skripts\poly_2_k.tol 5 Poly_2_Kug_OBJ
Poly_2_k.doc ;2d-Poly
```

```
ADD_SINGLE_TOOL_SKRIPT skripts\poly_2_k.tol 6 Poly_2_Kug_OBJ
Poly_2_k.doc ;3d-Poly
```

Dann weiß Reflections, daß man damit 2d- und 3d- Polygone „bearbeiten“ kann

Wichtiger Hinweis:

-Ihr Poly darf kein Leerzeichen im Namen haben, sonst funktioniert es nicht richtig. (Daran bin aber nicht ich schuld, ist schließlich bei den Standart -*.tol’s auch so)

-Ich bin nicht haftbar dafür, und kann nur hoffen, daß es bei Ihnen genauso funktioniert wie bei mir.

SET2ZERO

Was macht's:

Setzt alle Werte eines Objekts auf einer Achse (wird abgefragt) auf Null.

Wie installiert man es:

Ins Skript Verzeichnis kopieren und im Start Skript eintragen:

```
ADD_SINGLE_TOOL_SKRIPT skripts\set2zero.tol 3 set2zero  
set2zero.doc
```

Anmerkung:

Sinnvoll in Kombination mit dem panim-Skript.

Aber Achtung, das Ergebnis sieht aufgrund der sich nun überlappenden Polys nicht gut aus -> das Objekt sollte geeignet sein (z.B Landschaft wo es zu jedem x,y Wert nur genau einen z-Wert gibt)

OBJ2PRT

Diese Funktion erzeugt ein Partikelobjekt in der Form eines bestehenden Objektes.

Zuerst wird nach dem Objekt gefragt, welches nachgeahmt werden soll. Danach ob nur die selektierten Punkte behandelt werden sollen oder gleich alle. Dann den Namen des neuen Objektes.

Nun folgen immer Fragen über Minimum und Maximum der Partikel-Eigenschaften :

Stärke	-> Die Stärke der Partikel
Radius	-> Der Radius der Partikel
Rotanteil	-> Der Anteil roter Farbe (0-255)
Grünanteil	-> Der Anteil grüner Farbe (0-255)
Blauanteil	-> Der Anteil blauer Farbe (0-255)

Mit unterschiedlichen Werten der Maxima und Minima werden dann für jedes Partikel zufällige Werte dazwischen genommen.

Mit gleichen Werten für Maxima und Minima erhält jedes Partikel die selben Eigenschaften.

Die Parameter des Objektes können im Nachhinein nur mit dem Skript prtparam verändert werden.

Partikel müssen eine Stärke, sowie einen Radius besitzen ! 0 - Werte in beiden Fällen erzeugt Partikel die man weder im Editor noch nach dem Rendern sehen wird. Ohne Farbe werden die Partikel im Editor angezeigt doch auch beim Rendern führt dies nur zu unsichtbaren Punkten ...

VORSICHT mit grossen Objekten, da jedes Partikel einzeln gesetzt wird, kann es unter Umständen etwas länger dauern.

TEXTUREMAGIC 3D 1

TextureMagic 3D I - ab hier TM3D1 genannt - ist ein Texturplugin für das Raytracingprogramm "Reflections" ab 4.1.

Im Gegensatz zu Texturen, die auf Bilddateien basieren, berechnet TM3D1 die Texturen mit Hilfe von Algorithmen, die als Parameter die Punktkoordinaten und die vom Anwender eingestellten Werte erhalten.

Deswegen wird diese Art von Textur auch "mathematische Textur" oder besser "prozedurale Textur" genannt. In TM3D1 sind 20 verschiedenen Texturen enthalten:

ColorSpots
Vogonenkilt
ColorTracks
Schichtmarmor
Wolkenmarmor
Stufenmarmor
Nugget
Lochringe A
Lochringe B
Holiday
Bonbon
Ringworld
Hypno
Call It Love
Ringgitter
Magic Glass A
Magic Glass B
Verrottet
Asphalt
Hawaii



Beachten Sie bitte auch die allgemeinen Tips.

ColorSpots:

Diese Textur besteht aus Farbwolken.

Farben: 1

Parameter:

Intensität: Bestimmt die Veränderung der eingestellten Farbe.

Bei 0 entspricht die Textur einfach der Material-Grundeinstellung, bei 1 wird die Änderung maximal.

Chaos_X, Chaos_Y, Chaos_Z: Hier stellen Sie ein, wie stark die Textur verzerrt werden soll.

Hinweis:

Die Textur verwendet eine spezielle Art der Berechnung, bei der die Farben nicht immer wie gewohnt eingestellt werden können.

Vogonenkilt:

Das Muster auf dem Raumanzug von Prostetnik Vogon Jeltz.

Die Textur besteht aus Zellen, in denen sich Farbverläufe befinden.

Farben: 1

Parameter:

Intensität: Bestimmt die Veränderung der eingestellten Farbe.

Bei 0 entspricht die Textur einfach der Material-Grundeinstellung, bei 1 wird die Änderung maximal.

Chaos_X, Chaos_Y, Chaos_Z: Hier stellen Sie ein, wie stark die Textur verzerrt werden soll.

Ein weiteres Feature ist die mögliche "Übersteuerung" von Bereichen der Textur bei bestimmten Reglereinstellungen (Farbe/Intensität). Dadurch können zusätzliche Effekte erzielt werden.

Hinweis:

Die Textur verwendet eine spezielle Art der Berechnung, bei der die Farben nicht immer wie gewohnt eingestellt werden können.

ColorTracks:

Farbspuren

Farben: 1

Parameter:

Intensität: Bestimmt die Veränderung der eingestellten Farbe.

Bei 0 entspricht die Textur einfach der Material-Grundeinstellung, bei 1 wird die Änderung maximal.

Shift: Verschiebung und Verformung der Farbspuren.

Chaos_X, Chaos_Y, Chaos_Z: Hier stellen Sie ein, wie stark die Textur verzerrt werden soll.

Hinweis:

Die Textur verwendet eine spezielle Art der Berechnung, bei der die Farben nicht immer wie gewohnt eingestellt werden können.

Schichtmarmor:

Durch Zufallswerte verzerrte Farbschichten.

Farben: 4

Parameter:

Turbulenz: Verwirbelung der Schichten insgesamt.

Chaos_X, Chaos_Y, Chaos_Z: Hier stellen Sie ein, wie stark die Textur verzerrt werden soll.

Wolkenmarmor

Durch Zufallswerte verzerrte Farbschichten. Im Gegensatz zu Schichtmarmor[\[href=60\]](#) jedoch wolkig ineinander verlaufend.

Farben: 3

Parameter:

Chaos_X, Chaos_Y, Chaos_Z: Hier stellen Sie ein, wie stark die Textur verzerrt werden soll.

Stufenmarmor

Durch Zufallswerte verzerrte Farbschichten.

Auf den ersten Blick scheint sich Stufenmarmor nicht besonders von Schichtmarmor[\[href=60\]](#) zu unterscheiden. Das liegt daran, daß man den speziellen Unterschied im Vorschauenfenster nicht erkennen kann. Stufenmarmor besitzt einen Farbverlauf, bei dem die Farbveränderung der Ringe jeweils langsam ansteigt und dann plötzlich abbricht. Bei Schichtmarmor hingegen ist diese Ab- und Zunahme gleichmäßig.

Außerdem ist auch das Basismuster unterschiedlich.

Farben: 3

Parameter:

Chaos_X, Chaos_Y, Chaos_Z: Hier stellen Sie ein, wie stark die Textur verzerrt werden soll.

Nugget:

Goldtextur mit feinen Bumps.

Farben: 1

Parameter:

Bumps: Intensität des Bump-Effektes.

Lochringe A:

Verzerrten Ringe, die außer einem Farbverlauf auch transparente Stellen besitzen.

Farben: 2

Parameter:

Löcher: Die Breite der Spalten

Chaos_X, Chaos_Y, Chaos_Z: Hier stellen Sie ein, wie stark die Textur verzerrt werden soll.

Lochringe B:

Verzerrten Ringe, die außer einem Farbverlauf auch transparente Stellen besitzen. Anderes Muster als bei Lochringe A[\[href=100\]](#).

Farben: 2

Parameter:

Löcher: Die Breite der Spalten

Chaos_X, Chaos_Y, Chaos_Z: Hier stellen Sie ein, wie stark die Textur verzerrt werden soll.

Holiday:

Ineinander verlaufende Farbwolken.

Farben: 3

Parameter:

Chaos_X, Chaos_Y, Chaos_Z: Hier stellen Sie ein, wie stark die Textur verzerrt werden soll.

Bonbon:

Von einem Punkt ausgehende Farbstreifen.

Farben: 3

Parameter: -

Ringworld:

Kreisschichten.

Farben: 3

Parameter:

Dichte: Dichte der Schichten

Hypno:

Ein hypnotisierendes Muster aus 3 einstellbaren Farben.

Farben: 3

Parameter:

Dichte: Dichte des Musters.

Shift: Verschiebung des Musters.

Call It Love:

Nun, verwirrendes Muster, viele Schattierungen, die Voreinstellung ist rosa und spiegelt... ;-)

Farben: 6

Parameter:

Turbulenz: Verwirbelung der Schichten.

Chaos_X, Chaos_Y, Chaos_Z: Hier stellen Sie ein, wie stark die Textur verzerrt werden soll.

Ringgitter:

Einfaches Ringmuster mit einstellbaren transparenten Streifen.

Farben: 1

Parameter:

Gitterbreite: Verhältnis von Ringbreite zu Spaltenbreite.

Magic Glass A:

Farbverlauf, wobei der Startpunkt des Verlaufs die eingestellte Farbe und der Endpunkt "volle Transparenz" ist.

Farben: 1

Parameter:

Dichte: Feinheit des Musters.

Hinweise:

-Bei diesen Texturen ergeben sich oft interessante Schatten, die auch für Animationen verwendet werden könnten.

Magic Glass B:

Farbverlauf, wobei der Startpunkt des Verlaufs die eingestellte Farbe und der Endpunkt "volle Transparenz" ist. Für die transparenten Stellen kann eine zusätzliche Farbe gewählt werden.

Farben: 2

Parameter:

Dichte: Feinheit des Musters.

Hinweise:

-Bei diesen Texturen ergeben sich oft interessante Schatten, die auch für Animationen verwendet werden könnten.

Verrottet:

Meine Lieblingstextur aus TextureMagic3D I. Man kann sie für verrottetes Holz, Rost und noch vieles mehr verwenden. Testen Sie ruhig auch andere Farben, z.B. Blau und Weiß.

Farben: 2

Parameter:

Löcher: Hier stellen Sie ein, wie groß der Lochanteil sein soll.

Scramble: Wenn Ihnen die Musterverteilung nicht gefällt, wählen Sie einfach einen anderen Wert.

Randunschärfe: Scharfe Lochränder oder Auslaufen in steigender Transparenz

Asphalt:

Eine Textur mit einem sehr feinen Bumpmuster.

Farben: 1

Parameter:

Bumps: Intensität des Bump-Effektes.

Hinweis:

Für Asphalt u.ä. sollte "Bumps" niedrig stehen (wie in der Voreinstellung)

Hawaii:

Gehen Sie eine Runde surfen...

Animierbares Wellenmuster.

Farben: 2

Parameter

Wellenhöhe: Sturm oder spiegelglatte See?

Dichte: Dichte der Wellen.

Shift: Bewegung des Musters.

Chaos_X, Chaos_Y, Chaos_Z: Hier stellen Sie ein, wie stark die Textur verzerrt werden soll.

Hinweis:

Wichtig ist bei der Animation der Parameter "Shift", der die Wellen bewegt. Beachten Sie bitte, daß das Zentrum der Welle immer das Zentrum des lokalen Koordinatensystems des Texturobjektes ist (das ist die Ecke, an der "0" steht).

Wenn die Wellen also beispielsweise in der Mitte einer Fläche beginnen sollen, müssen Sie das Texturobjekt mit dem Nullpunkt an diese Stelle verschieben (in 2 Ansichten).

Tips

Allgemein:

-Falls Sie auf irgendwelche Unregelmäßigkeiten stoßen sollten, sind dies in der Hilfe nicht erwähnte Features.

Farbeinstellung:

-Wenn bei einer Textur mehrere Farben eingestellt werden können, können Sie sich über die Farbverteilung einen Überblick verschaffen, in dem Sie für jeweils eine Farbe Werte einstellen, die sich stark von den übrigen Farben unterscheiden (z.B. diese Farbe weiß, alle anderen schwarz). So sehen Sie sofort, welche Position und Gewichtung diese Farbe hat.

Skalierung, Rotation, Verschiebung:

-Diese Operationen werden über das jeweilige Texturobjekt durchgeführt, nicht durch die Texturparameter selbst.

Animation:

-Alle Parameter, die sie von Hand einstellen können, können Sie auch von Reflections animieren lassen (siehe Handbuch). Auch das Textur-objekt kann animiert werden. Besonders interessant können die Shift-Parameter sein, die bei einigen Texturen vorhanden sind.

Sonstiges:

-Wenn Sie bei einer teiltransparenten Textur durch die Löcher die Rückseite des Objektes sehen wollen, verwenden sie bitte

1. bei Ref 4.1 keine mathematischen Kugeln.
2. eine angemessene Raytracingtiefe

Für alle Texturen gilt: Man kann sie oft für viel mehr Dinge einsetzen, als ihr Name vermuten läßt. Sie sollten also ruhig etwas experimentieren. Sie sollten darauf achten, daß der durch Bumps hervorgerufene Kontrast nicht zu hoch ist. Also in diesen Fällen den entsprechenden Regler etwas zurückstellen.

Installation:

-Falls Sie die Reflections 4.2 CD-ROM besitzen, sollte dieses Plugin automatisch installiert werden. Falls es dennoch irgendwelche Probleme gibt oder Sie das Plugin von Hand nachinstallieren möchten:

Die Dateinamen auf der CD-ROM haben kurze Namen (8.3). Sie müssen die Dateien folgendermaßen kopieren und umbenennen:

```
tm3dI.dll    -> <ref-verzeichnis>\plugin\mtex\TextureMagic 3D I.dll
tm3dI.inf    -> <ref-verzeichnis>\plugin\mtex\TextureMagic 3D I.inf
tm3dI.hlp    -> <ref-verzeichnis>\help
```

Das Zielverzeichnis der Hilfedatei können Sie sich allerdings aussuchen, da sie ohnehin von Hand aufgerufen werden muß. Wichtig ist, daß die DLL und die INF-Datei bis auf die Dateierweiterung "dll" bzw. "inf" den genau gleichen Namen haben, da sonst das Plugin nicht im Reflections-Menü auftauchen wird.

LANDSCAPE

Nur AMIGA.

Landscape ermöglicht das erstellen von fraktalen Landschaften für Reflections ab dem Dateiformat der Version 4.0.

Oder auch als DXF zum importieren. Ebenfalls kann eine Höhenkarte im TGA Format abgespeichert werden.

Dem Programm beigelegt ist noch eine Ausführliche AmigaGuide - Datei. Wohin Sie alles Installieren ist frei, am besten zum Beispiel in ein Verzeichnis das im Pfad liegt. "path <verzeichnis> add"

Somit können Sie das Script von überall aus Starten.

Ebenfalls ein ARexx-Skript ist dabei um die Bedienung von Reflections aus zu erleichtern. "landscape.ref_rexx"

Somit ist es möglich direkt von Reflections aus eine Landschaft zu erstellen und in die Szene einzubinden.

Dazu sollte dieses Skript inklusive dem landscape Programm in das <reflections>/ARexx Verzeichnis kopiert werden.

Womit Sie es via Kontrolle>Landscape aufstarten können.

Jetzt sind noch die zwei Variablen : TempDir -> Wo die landschaften temporär zwischengespeichert werden, ohne letzten Slash.

Aber nicht gelöscht !! Reflections mach einen exklusiven Lock darauf.

LandScapePath -> Pfad und Name des programmes.Der Landschaftsgenerator selbst hat folgende Parameter:

F=Filename/A,X=XDelta/K/N

S=Steps/K/N

H=Height/K/N

R=RandomSeed/K/N

DXF=SaveDXF=/S

TGA=SaveTGA=/S

REF41=SaveREF41/S

V=Verbose/S

Filename : Dateiname in welche die Landschaft gespeichert werden soll. !! OHNE Endung, diese wird je nach Dateityp selbst erstellt.

XDelta : Auflösung der Matrix.

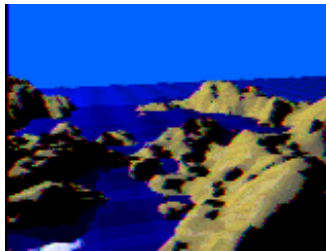
BSP : alle 25 Einheiten einen Punkt.

Steps : Anzahl Berechnungsschritte. 2^{Steps} = Kantenlänge.

Mit Steps=8 erhält man also ein Feld von 256×256 .

Height : Maximale Höhe(Tiefe) in der Landschaft.

RandomSeed : Initialer Zufallswert.
 SaxeDXF : Als DXF Datei speichern. Braucht viel Platz !!
 SaveTGA : Als TGA Graustufenbild speichern (Höhenkarte).
 SaveREF41 : Als Reflections .r4 Datei speichern.
 Verbose : Keine Berechnung wird vorgenommen, lediglich
 die gewählten Argumente werden Angezeigt.
 Die Berechnung kann jederzeit mit CTRL-C abgebrochen werden.
 Mit "landscape test REF41 DXF TGA" wird das Resultat in allen drei
 Dateitypen ausgegeben !
 Bemerkung : Die Standartargumente sind X=1 S=1 H=10 R=100 somit
 wird mit "landscape test REF41" eine Datei "test.r4" angelegt mit 4 Punk-
 ten von maximaler
 Höhe=10 und Abstand=1 und einem Initialzufallswert=100
 Weitere Speicherformate, auch Reflections < 4.0, und auch bessere Bild-
 formate sind in vorbereitung.
 Sowie die Grafische Oberfläche.



PRTPARAM

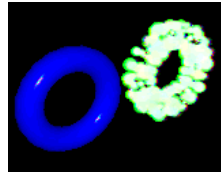
Diese Funktion ermöglicht es die Parameter eines Partikelobjektes zu ändern, das zum Beispiel mit obj2prt erstellt wurde.

Zuerst wird nach dem Partikelobjekt gefragt. Dann ob nur die selektierten Punkte behandelt werden sollen, oder das gesamte Objekt.

Jetzt bleibt noch die Wahl zwischen Farbe und Größe die verändert werden soll.

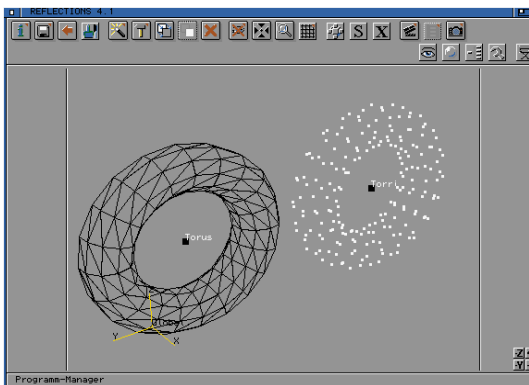
Nun folgen immer Fragen über Minimum und Maximum der Partikeleigenschaften :

- Stärke -> Die Stärke der Partikel
- Radius -> Der Radius der Partikel
- Rotanteil -> Der Anteil roter Farbe (0-255)
- Grünanteil -> Der Anteil grüner Farbe (0-255)
- Blauanteil -> Der Anteil blauer Farbe



Mit unterschiedlichen Werten der Maxima und Minima werden dann für jedes Partikel zufällige Werte dazwischen genommen.

Mit gleichen Werten für Maxima und Minima erhält jedes Partikel die selben Eigenschaften.



©REFTools!team/Stephan Biemann

Das Reftools Team



Name: Christian Hendorfer
geb.: k.A.
Nickname: Henne
IRC-Nick: CHH
E-Mail: chh@odn.de



Name: Clemens Hochreiter
geb.: 27.12.79
Nickname: Pyromite
IRC-Nick: Pyro
E-Mail: Clemens.Hochreiter@munich.netsurf.de



Name: Danilo Reinhardt
geb.: 22.01.78
Nickname: Dani
IRC-Nick: LS,Bossman,Psycho
E-Mail: Lightshine@gmx.net, Bossman@gmx.net

Das Reftools Team



Name: Dirk Kamp
geb.: 5.02.1969
Nickname: El Presidente
IRC-Nick: SEG
E-Mail: dirkkamp@cww.de
URL: <http://privat.schlund.de/reftools>



Name: Dominik Fenzl
geb.: 1982
Nickname: Dom
IRC-Nick: fM4
E-Mail: fM4@gmx.net
URL: <http://good.home.pages.de/>
ICQ-UIN: 3994654



Name: Cem Capin
geb.: k.A.
Nickname: k.A.
IRC-Nick: k.A.
E-Mail: Capin@i-bs.de

Das Reftools Team



Name: Klaus-Peter Schreiner
 geb.: k.A.
 Nickname: Juli
 IRC-Nick: Juli,Peter
 E-Mail: kpeter.schreiner@trans-it.de
 URL: <http://home.t-online.de/home/klaus-peter.schreiner/reftools.htm>



Name: Lars Iwer
 geb.: 14.09.80
 Nickname: Argon
 IRC-Nick: Argon
 E-Mail: reflections@cwv.de
 URL: <http://dimx.eucenter.com>



Name: Magnus Möller
 geb.: 19.07.69
 Nickname: Maggi Maniac
 IRC-Nick: IceDog
 E-Mail: mmoeller@on.luebeck.de
 URL: <http://www.on-luebeck.de/~mmoeller>

Das Reftools Team

kein
Foto

Name: Michael Alexander Krafzik
geb.: k.A.
Nickname: Slowhand
IRC-Nick: Slowhand;(_mak)
E-Mail: michael.krafzik@duisburg.netsurf.de



Name: Michael Bischof
geb.: 9.3.79
Nickname: bischy
IRC-Nick: bischy
E-Mail: Michael.Bischof@t-online.de
URL: <http://home.t-online.de/home/Michael.Bischof/index.htm>



Name: Philipp Dortmann
geb.: 6.9.75
Nickname: k.A.
IRC-Nick: Gastar
E-Mail: gastar@t-online.de

Das Reftools Team



Name: Ricardo Brucksch
geb.: 3.1.74
Nickname: igusto
IRC-Nick: igusto
E-Mail: igusto@t-online.de
URL: <http://dimx.eucenter.com>



Name: Stephan Bielmann
geb.: k.A.
Nickname: k.A.
IRC-Nick: Stean
E-Mail: stephan.bielmann@rega-sense.ch
URL: <http://www.rega-sense.ch/~sbielmann/>



Name: Thomas Hendorfer
geb.: k.A.
Nickname: Heni
IRC-Nick: tuk
E-Mail: tuk@odn.de