



www.Amanda3D.de
Die dicke fette Tante im Netz!

Einführung: "Thinking Partices" für Maxon Cinema 4D

Einführung : "Thinking Particles"

für Maxon Cinema 4D ab Release 8

(Stand 06/05)

Alle Rechte vorbehalten. Dieses Tutorial und die dazugehörigen Dateien sind urheberrechtlich geschützt. In diesem Tutorial steckt eine Menge Arbeit. Daher darf es ohne vorheriger, ausdrücklicher und schriftlicher Genehmigung in keiner Form (auch nicht auszugsweise) mittels irgendwelcher Verfahren reproduziert, gesendet, geändert, vervielfältigt oder verbreitet werden.

Auch, wenn darauf geachtet wurde, ein fehlerfreies Tutorial zu erzeugen, können Fehlbeschreibungen nicht ausgeschlossen werden. Für Schäden jeglicher Art, wie z.B. Schäden an Hard oder Software oder den Verlust von Daten, die auf eine Fehlbeschreibung des Tutorials oder auf Fehler in den mitgelieferten Dateien zurückzuführen sind, wird keine Haftung übernommen.

Inhaltsverzeichnis

1 Partikel Emitter	3
2 Emitter zeitlich steuern	6
3 Partikelobjekte	9
4 Gravitationsfelder	13
5 Reflektoren	18
6 Partikel folgen Objekten	21
7 Objekte als Partikelemitter	24
8 Objekte als Volumenemitter	29
P1 Projekt "Schneekugel"	33

1 | Partikel Emitter

Mit „Thinking Particles“ wurde Cinema 4D um ein sehr flexibles und komplexes Partikelsystem erweitert, welches eine absolute Kontrolle der Partikel gewährleistet und somit Partikelanimationen ermöglicht, die man vorher mit Standardmitteln nicht für möglich gehalten hätte.

„Thinking Particles“ wird über den XPresso-Editor und den Attribut-Manager gesteuert.

Auf den ersten Blick wirkt „Thinking Particles“ etwas verwirrend. Um den Einstieg in das Modul zu erleichtern, habe ich neun Tutorials geschrieben, die vom Schwierigkeitsgrad ansteigend sind.

Es beginnt mit dem einfachen „Partikel emittieren“ und endet mit der Simulation einer „Schneekugel“, welche man bei entsprechender Hardware sogar schütteln kann.

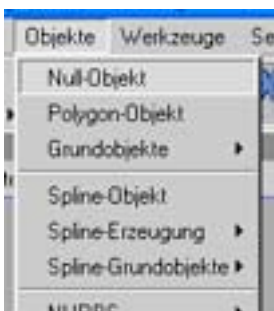
Wir fangen an:

Wo finde ich „Thinking Particles“ und wie bringe ich es dazu Partikel zu „sprühen“ ?

Erstmal solltet Ihr eine neue Szene erzeugen.

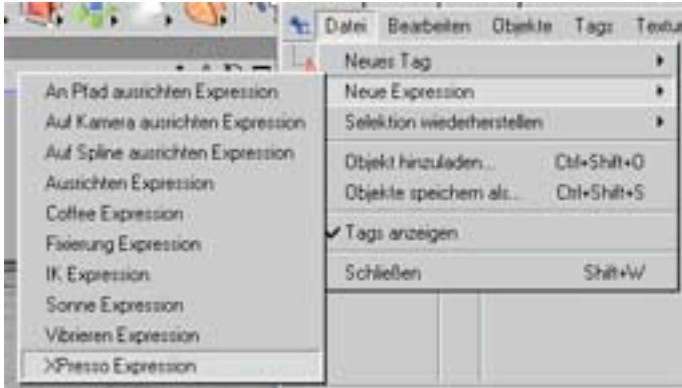
Da „Thinking Particles“ hauptsächlich über den XPresso-Editor gesteuert wird, muss ein Objekt in der Szene erstellt werden, das eine XPression aufnehmen kann. Hierfür drängt sich ein Null-Objekt förmlich auf, da es nicht sichtbar ist und keine Polygone enthält.

Objekte-> Null-Objekt



Markiert im Objekt-Manager das Null-Objekt und fügt diesem eine neue Expression zu.

Objekt-Manager->Datei->Neue XPression->XPresso Expression



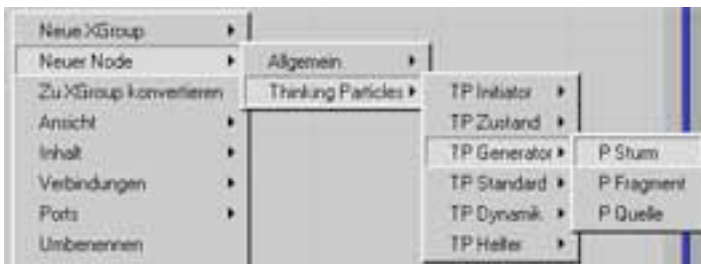
Neben dem Null-Objekt erscheint ein Symbol . Gleichzeitig springt der XPression-Editor auf. Das Symbol zeigt an, dass eine "XPression" existiert. Sollte der Editor mal nicht sichtbar sein, könnt Ihr ihn jederzeit mittels Doppelklick auf das Symbol öffnen.

Neben ein paar Sonderfällen gibt es drei "Hauptemitterarten" in "Thinking Particles" :

P Sturm, P Fragment und P Quelle

Wenn Ihr mit der rechten Maustaste in einen freien Bereich des XPresso-Editors klickt, öffnet sich ein Kontext Menü. Wählt hier PSturm aus.

Rechte Maustaste->Neuer Node->Thinking Particles->TP Generator->PSturm



Nun habt Ihr einen Emitter erzeugt. Ein Emitter ist für das Aussenden von Partikeln zuständig.

Standardmäßig erscheint jetzt als Symbol ein weißer Kreis in der 3D Ansicht. Wenn Ihr nun auf die Abspieltaste drückt, fängt der Kreis an, Partikel auszusenden.

Klickt nun im XPresso-Editor auf die Titelzeile des PSturm-Nodes, dieser wird rot umrandet. Im Attribut-Manager könnt Ihr diverse Einstellungen wie Form, Art des Aussendens, Geschwindigkeit, Richtung und vieles mehr vornehmen.

Wahrscheinlich werdet Ihr nun feststellen, dass man den Emitter weder verschieben, noch drehen kann. Damit das möglich ist, müsst Ihr das zuvor erstellte Null-Objekt mit dem Emitter verknüpfen.

Schön und gut – aber – wie geht das ? Ganz simpel ! Zieht einfach das Null-Objekt aus dem Objekt Manager in den XPresso-Editor.

Damit Ihr die beiden Nodes (So heißen die Kästchen...) miteinander verbinden könnt, müsst ihr noch für entsprechende Ein- und Ausgangspunkte sorgen. Eingänge kommen immer nach links (blaue Seite) - die Ausgänge nach rechts (rote Seite).

Um einen Ein- und Ausgang zu erstellen, klickt Ihr mit der linken Maustaste auf das jeweilige Quadrat (blau oder rot) in der Titelzeile des Nodes. Ein Dropdownmenü erscheint.

Erstellt nun folgende Ein- und Ausgänge :

Null-Objekt :

Ausgang : Globale Matrix

Koordinaten->Globale Position-> Globale Position

PSturm :

Eingang : Emitter Ausrichtung

Zieht nun Linien von den Ausgangspunkten des Null-Objekt Nodes zu den Eingangspunkten des PSturm Nodes (siehe Bild).



Die Globale Position teilt dem Emitter mit, an welcher Stelle im 3D Raum er sich befinden soll (nämlich da, wo das Null-Objekt ist...) Die Globale Matrix richtet den Emitter PSturm nach dem Null-Objekt aus.

Probiert es aus. Drückt die Abspieltaste und verschiebt und dreht das Null-Objekt. Der Emitter folgt automatisch.

Fertig! Ihr seid nun in der Lage, einen einfachen „Thinking Particles“ Emitter in Eure Szenen einzubauen.

Im zweiten Tutorial werde ich Euch zeigen, wie Ihr den Partikel Emitter nur für bestimmte Zeiträume aktiviert.

2 | Emitter zeitlich steuern


Im vorliegenden Tutorial beschäftigen wir uns damit, wie ein Partikel-Emitter dazu gebracht werden kann, Partikel nur zu festgelegten Zeiten auszustoßen.

Da der grobe Ablauf für die Erstellung von Partikelsystemen fast immer gleich ist, können wir getrost die Scene des ersten Tutorials übernehmen : tp_tut1(fertig).c4d

Wenn Ihr auf die Abspieltaste drückt, werdet Ihr feststellen, dass der Emitter vom Frame 0 bis in alle Ewigkeiten Partikel aussendet.

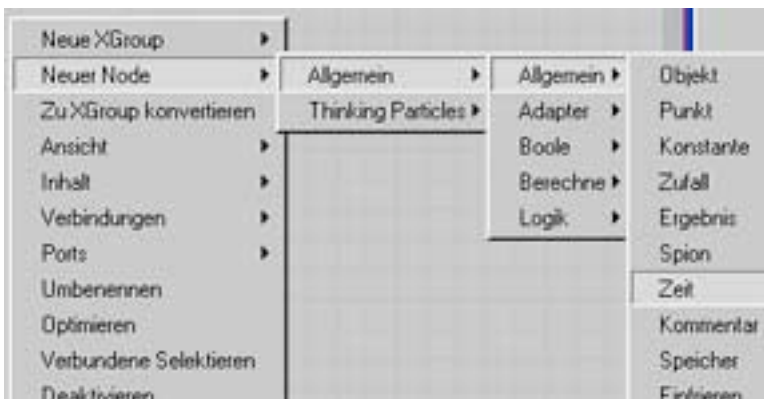
Um der Sache Herr zu werden, gibt es verschiedene Möglichkeiten.

Wir fangen daher mit der einfachsten Variante an:

Öffnet den XPresso Editor durch einen Doppelklick auf das entsprechende Symbol im Objekt-Manager .

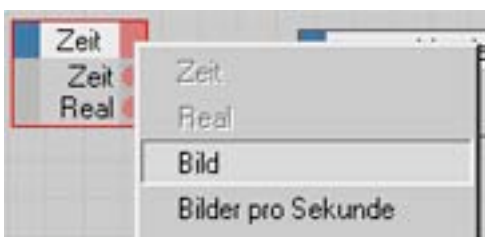
Erstellt im XPresso-Editor ein Zeit-Node. Ein Zeit-Node liefert die Zeit einer Animation.

Rechte Maustaste->Neuer Node->Allgemein->Allgemein-> Zeit



Im vorliegenden Fall benötigen wir eine Angabe der Zeit in Bildern. Daher benötigen wir im Zeit-Node einen Ausgang, der die aktuelle Bildnummer ausgibt. Erstellt nun einen Bild Ausgang.

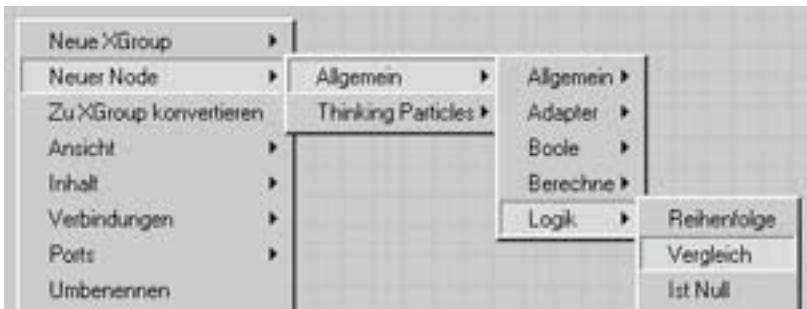
Linke Maustaste auf das rote Feld->Bild



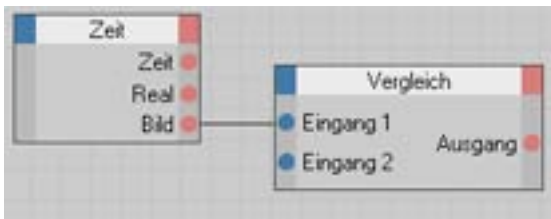
Der Zeit-Node allein macht aber keinen Sinn. Ihr müsst dem Emitter natürlich noch mitteilen, in welchem Zeitraum Partikel ausgesendet werden sollen.

Hier wird also wieder ein Node benötigt. Das Vergleichs-Node hier die Lösung.

Rechte Maustaste->Neuer Node->Allgemein->Logik->Vergleich



Ein Vergleichs-Node kann, wie der Name schon sagt, Werte miteinander vergleichen. Damit das Vergleichs-Node weiß, „wieviel Uhr es ist“, müsst Ihr nun den Bild Ausgang des Zeit-Nodes mit dem Eingang 1 des Vergleichs-Nodes verbinden.



So! Auf die Uhr gucken kann das Node nun schon einmal. Bringt ihm aber nicht viel, da er nichts mit den Daten anfangen kann.

Das müsst Ihr ihm schon beibringen. Markiert hierzu das Vergleichs-Node. Im Attribut-Editor müsst Ihr noch kleiner/gleich (\leq) und für Eingang 2 den Betrag „50“ eintragen.

Der Vergleichs-Node prüft nun, ob die aktuelle Bildnummer unter oder gleich 50 liegt.

Wenn Ihr nun auf Abspielen drückt, passiert leider immer noch nichts.

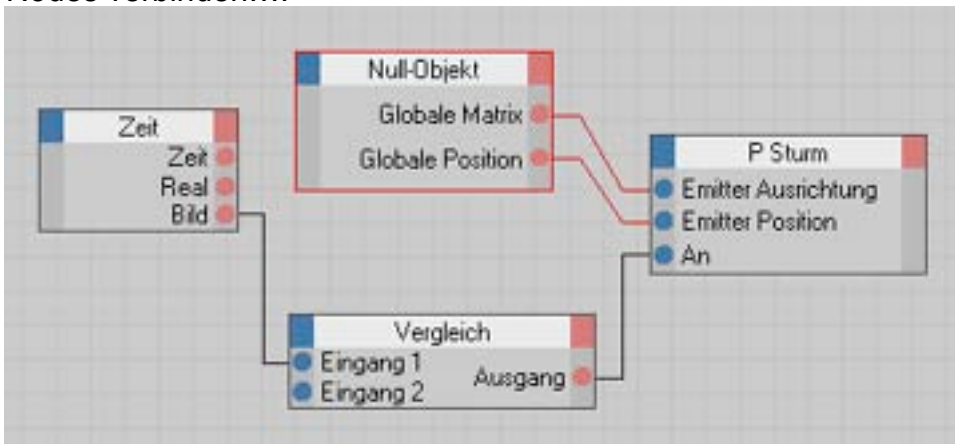
Woran liegt das ?

Wenn Ihr die Verknüpfungen der Nodes im XPresso-Editor anschaut, werdet Ihr recht schnell sehen, dass noch keine Verbindung zum Emitter existiert.

Das holen wir nun schnell nach. Erstellt im PSturm-Node einen „An“ Eingang

Linke Maustaste auf das blaues Feld->An

Nun müsst Ihr nur noch den Ausgang des Vergleichs Nodes mit dem Eingang des PSturm Nodes verbinden....



...und die Abspieltaste drücken. Die Partikel werden jetzt nur bis Bild 50 ausgesandt. Danach werden keine neuen mehr geboren.

So das war alles. Es gibt noch verschiedene Möglichkeiten, Partikel zeitlich zu beeinflussen. Zum Beispiel durch Intervalle (Partikel ausstoss für bestimmte, sich wiederholende Zeiträume).

Da wir aber hier nur die absoluten Basisinge besprechen wollen, hebe ich mir das für das projektorientierten Tutorial auf.

Als nächstes werden wir Objekte als Partikel auf die Reise schicken.

3 | Partikelobjekte

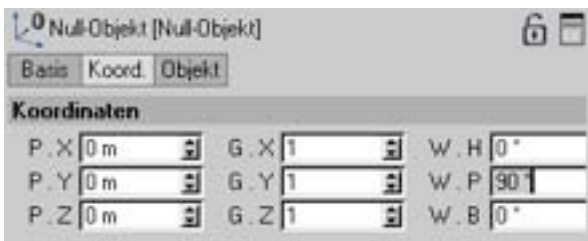
Und schon sind wir bei Teil 3 der Basis-Tutorials. Wie in jedem Partikelsystem kann man natürlich auch Objekte als Partikel verwenden.

Das Tutorial ist diesmal etwas kürzer.

Schon geht's los:

Erstellt ein Basis-Partikelsystem, wie Ihr es im ersten Tutorial gemacht habt. Alternativ könnt Ihr auch mit mitgelieferte Szene verwenden: **tp_tut01(fertig).c4d**

Dreht den Emitter bzw. das Nullobjekt um 90 Grad gegen den Uhrzeigersinn. Wenn Ihr die Abspieltaste drückt, dann müssten die Partikel nun nach „oben“ fliegen



Objekte als Partikelobjekte zu verwenden ist recht einfach.

Zunächst müsst Ihr ein Objekt erstellen, welches als Partikel-Objekt dienen soll. Ich habe mich hier für eine einfache Kugel mit 10m Durchmesser entschieden.

Da der P Sturm-Node über den Abmessung-Wert das Objekt (in diesem Fall die Kugel) skaliert, sollte dieser Wert des P Sturm-Nodes nach Möglichkeit auch auf „10“ gestellt werden.

Öffnet hierzu den Xpresso-Editor mittels Doppelklick auf das entsprechende Symbol im

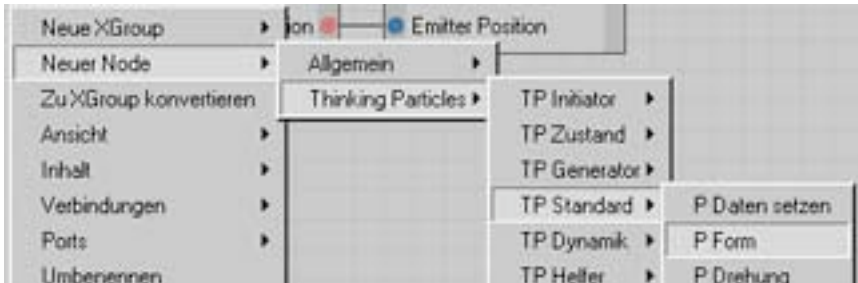


Objekt-Manager () und markiert den P Sturm-Node. Im Attribute-Manager könnt Ihr den Wert nun einstellen.

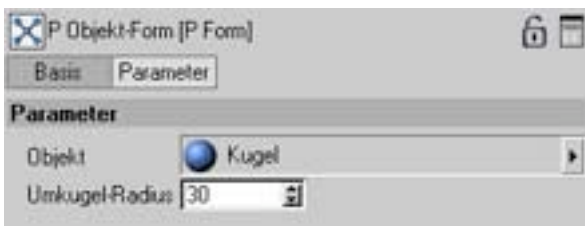


Danach müsst Ihr im XPresso-Editor ein „P Form-Node“ aufrufen.

Re.Maustaste->Neuer Node->Thinking Particles-TP Standard->P Form



Der P Form-Node ist dafür Zuständig, um dem Emitter zu sagen, welche Form die Partikel haben sollen. Übergebt nun das erstellte Kugelobjekt an das P Form-Node, indem Ihr die Kugel mit gedrückter linker Maustaste aus dem Objekt-Manager in das Objekt-Feld des P Form-Nodes im Attribute-Manager zieht und auch hier als Umkugelradius den Wert „10“ einträgt.

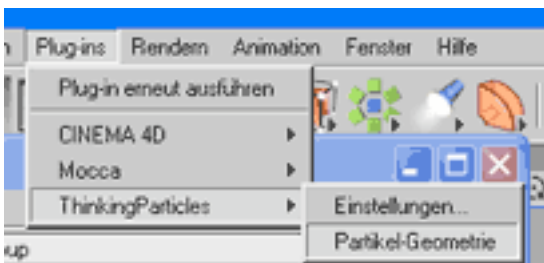


Wie kann der P Form-Node dem Emitter denn sagen, welche Form die Partikel haben sollen, wenn er das selbst nicht weiß ?

Nun das werden wir ihm nun mitteilen!

Dazu müsst Ihr ein Partikel-Geometrie Objekt erstellen.

Plugins->Thinking Particles->Partikel-Geometrie

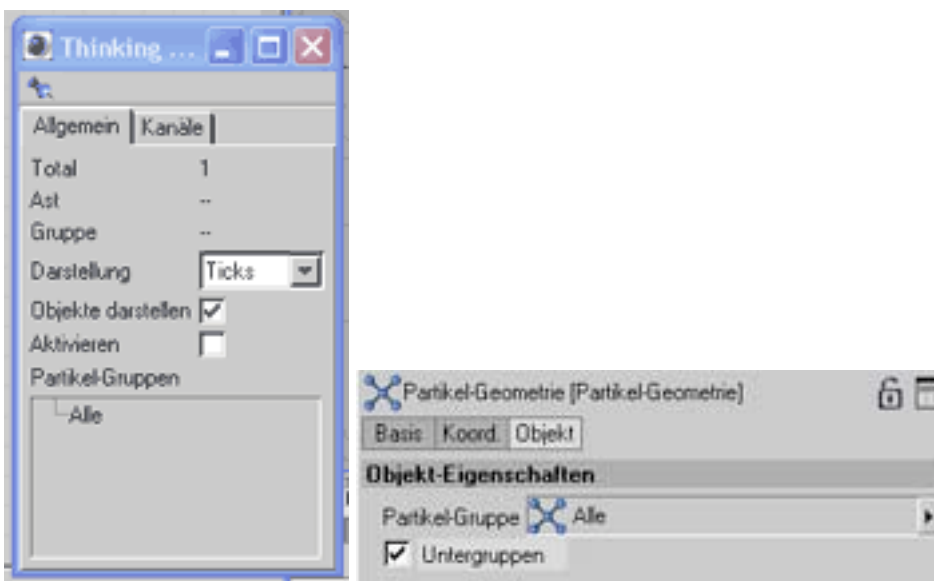


Wenn das neu erstellte Partikel-Geometrie Objekt im Objekt-Manager markiert ist, seht Ihr im Attribute-Manager ein Feld Namens „Partikel-Gruppe“. Über Partikelgruppen wollte ich im Moment noch nicht reden. Nehmt daher einfach folgende Einstellung vor:

XPresso-Editor-Zusätze->Thinking Particles->Einstellungen

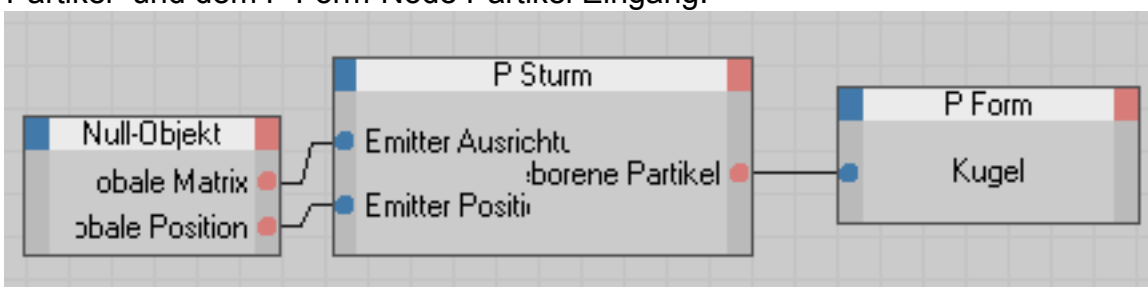


Ein Fenster für die Einstellungen von "Thinking Particles" öffnet sich. Im unteren Bereich gibt es ein Feld, welches den Namen "Partikel-Gruppen" enthält. In diesem Feld sind alle bisher erstellten Partikelgruppen enthalten. Im vorliegenden Fall dürfte dort „Alle“ stehen. Zieht mit der gedrückten linken Maustaste das Wort „Alle“ in das Partikel-Gruppe Feld des Partikel-Geometrie Objektes im Attribute-Manager.



Das Eigenschaften Fenster könnt Ihr wieder schließen.

Nun müsst Ihr noch eine Verbindung zwischen dem Emitter und dem P Form-Node herstellen. Zieht daher eine Linie zwischen dem P Sturm-Node (Emitter) Ausgang „Geborene Partikel“ und dem P Form-Node Partikel Eingang.



Drückt nun auf die Abspieltaste wenn Ihr alles richtig gemacht habt, müssten nun ganz viele kleine Kügelchen "ausgespuckt" werden.

Um ein wenig Ordnung in Eurer Szene zu machen, solltet Ihr das eigentliche Kugelobjekt unsichtbar machen und dem Partikel-Geometrie Objekt unterordnen. Damit behaltet Ihr auch bei größeren Szenen leichter den Überblick!

Das Tutorial wäre damit zu ende!

Im nächsten Tutorial werden wir uns um Gravitationsfelder kümmern.

4 | Gravitationsfelder

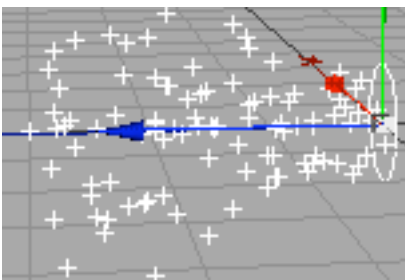
Sobald die Partikel den Emitter verlassen haben, fehlt uns jegliche Kontrolle. Bisher haben wir ausschließlich den Emitter beeinflusst, um die Partikel unseren Wünschen anzupassen.

„Thinking Particles“ stellt eine Reihe von Möglichkeiten zur Verfügung, um das Verhalten der Partikel auch dann kontrollieren zu können, nachdem sie den Emitter verlassen haben.

In diesem Tutorial werden wir die Partikel sogenannten Gravitationsfeldern (Kraftfeldern) aussetzen.

Erstellt erneut ein Partikel Grundsystem, wie im Basis Tutorial 1 beschrieben, oder verwendet die mitgelieferte Datei: ***tp_tut01(fertig).c4d***

Wenn Ihr auf die Abspieltaste drückt, dann fliegen die Partikel seitlich weg.



Wir möchten nun aber einen Wasserfall oder einen Wasserstrahl aus einem Gartenschlauch simulieren. In solchen Fällen fließt das Wasser zunächst geradeaus, um dann in einem Bogen nach „unten“ zu rauschen.

Solch eine Situation kann man zum Beispiel mittels sogenannter Gravitationsfelder / Kraftfelder realisieren.


„Thinking Particles“ stellt verschiedene Arten von Kraftfeldern zur Verfügung (z.B. Kugel und Fläche).

In unserem Fall benötigen wir ein Gravitationsfeld Fläche, um die Partikel zum Fallen zu bringen.

Weiterhin werden zum ersten mal die Partikelgruppen zum Einsatz kommen. Partikelgruppen sind ein elementarer Bestandteil von „Thinking Particles“. Sie ermöglichen überhaupt erst komplexe Partikelanimationen.

Jeder Partikel gehört einer Partikelgruppe an. Durch die Zugehörigkeit von Partikelgruppen ist es möglich, Partikel einer bestimmten Gruppe von den Partikeln und deren Verhalten einer anderen Partikelgruppe abzugrenzen. Ein kontrollierter Wechsel der Partikel von einer Gruppe zur nächsten ist problemlos möglich.

Auch, wenn Ihr das jetzt nicht genau verstanden habt, wird es innerhalb der nächsten Tutorials nachvollziehbar.

Öffnet zunächst den Xpresso-Editor durch einen Doppelklick auf das entsprechende Symbol im Objekt-Manager ()

Im XPresso-Editor ruft Ihr die Thinking Particles Einstellungen auf :

XPresso-Editor->Zusätze->Thinking Particles->Einstellungen



In diesem Fenster seht Ihr wieder im Partikel-Gruppe Feld das Wort "Alle". Das Wort stellt schon eine Partikelgruppe dar. Diese Gruppe ist immer vorhanden und ist nicht löschar. Sämtliche Partikel, die keiner besonderen Partikel-Gruppe zugeordnet sind, gehören zur „Alle“ Gruppe.

Es ist nun an der Zeit, eine neue Gruppe zu erstellen:

„Alle“ markieren->rechte Maustaste->Hinzufügen

Der „Alle“ Partikelgruppe wird nun eine neue Partikelgruppe untergeordnet. Durch Doppelklick auf den neuen Gruppennamen kann die Gruppe umbenannt werden (hier „meinePartikel“). Hiervon solltet Ihr unbedingt Gebrauch machen, da nur eine individuelle Namensgebung bei größeren Projekten einen besseren Überblick garantiert.

Der „meinePartikel“-Gruppe muss nun noch mitgeteilt werden, für welche Partikel sie zuständig ist.

Damit man einer Partikelgruppe Partikel zuordnen kann, gibt es sinnigerweise ein „P Gruppe“-Node, dass Ihr im XPresso-Editor erstellen müsst.

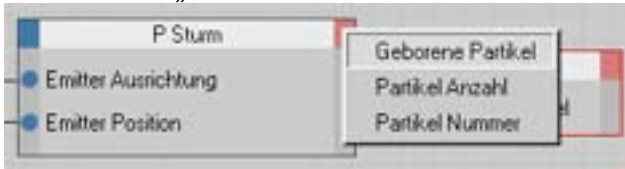
Re.Maustaste->Neuer Node->Thinking Particles->TP Standard->P Gruppe

Markiert im XPresso-Editor den neuen Node (rot umrandet). Im Attribute-Editor erscheint unter Parameter ein Feld für die Partikelgruppe. Zieht nun aus dem noch geöffneten „Thinking Particles“ Einstellungsfenster die „meinePartikel“-Gruppe mit der gedrückten linken Maustaste in dieses Feld.



Nun müsst Ihr nur noch den Eingang des P Gruppe mit dem Ausgang vom „P Sturm“-Node verbinden.

Erzeugt hierzu einen Ausgang beim „P Sturm“-Node mit dem Namen geborene Partikel (falls nicht schon vorhanden) mittels Mausklick auf das rote Ausgangskästchen in der Titelzeile des „P Sturm“-Nodes..

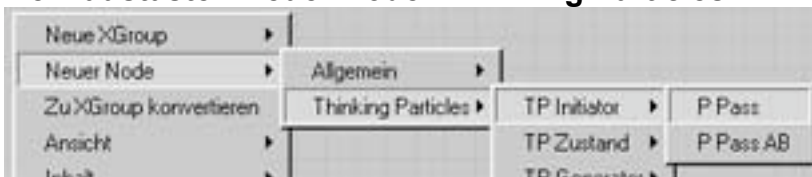


Zieht nun eine Linie von diesem Ausgang zum Eingang des „P Gruppe“-Nodes.

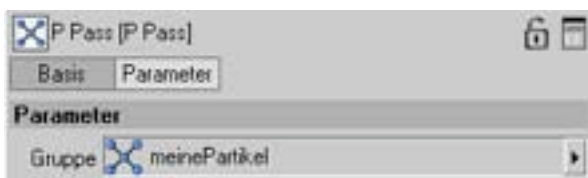
Geschafft! Sämtliche Partikel, die jetzt durch den „P Sturm“-Node geboren werden, sind der „meinePartikel“-Gruppe zugeordnet.

Ihr werdet euch nun fragen, wie man den nun wieder auf diese Partikel zugreifen kann. Jawoll, auch daran wurde natürlich gedacht. Hierfür ist der „P Pass“-Node vorgesehen. Und genau den wollen wir jetzt erzeugen:

Re.Maustaste->Neuer Node->Thinking Particles->TP Initiator->P Pass



Auch hier muß wieder eine Partikelgruppe übergeben werden. Öffnet wieder das „Thinking Particles“ Einstellungen Fenster, markiert den „P Pass“-Node. Und zieht wieder den Partikelgruppennamen in das hierfür vorgesehene Gruppe Feld des „P Pass“-Nodes im Attribute-Manager.



Dieser Node stellt nun sämtliche Partikel der jeweiligen Partikelgruppe am Ausgang zur Verfügung.

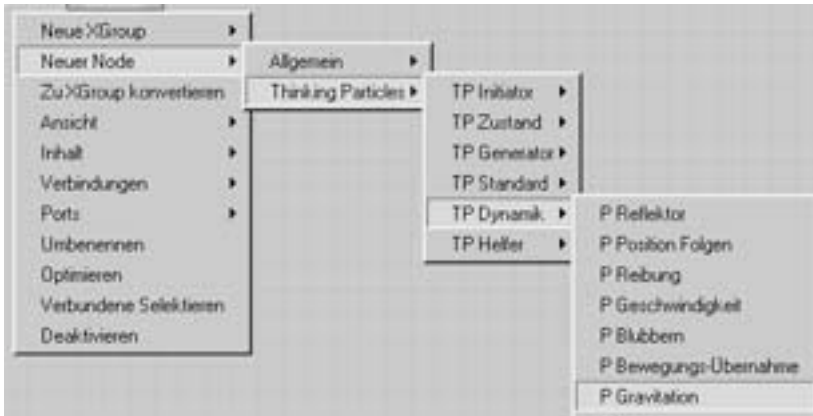
OK, aber wofür brauchen wir das jetzt ?

Wir hatten vor, ein Flächen-Gravitationsfeld zu erzeugen, welches die Partikel dazu bringt nach unten zu fallen. Das Gravitationsfeld will natürlich wissen, welche Partikel angezogen oder abgestoßen werden sollen. Diese Information muss grundsätzlich mit einem „P Pass“-Node erteilt werden.

Weil es so schön ist, erstellen wir noch ein Node. Den „P Gravitation“-Node. Was dieser Node machen soll brauche ich ja nicht mehr erklären.

Auch dass „P Gravitations“-Node muß an ein Objekt gekoppelt sein. Wie immer empfiehlt sich hier wieder ein Null-Objekt. Bitte nennt dieses „Gravitation“.

Re.Maustaste->Neuer Node->Thinking Particles->TP Dynamik->P Gravitation



Anschließend müßt Ihr das „Gravitations-Null-Objekt“ noch dem „P Gravitation“-Node zuweisen, um eine Verbindung herzustellen. Zieht daher das „Gravitation“-Objekt vom Objektinspektor in das Objekt-Feld von „P Gravitation“ im Attribute-Manager.



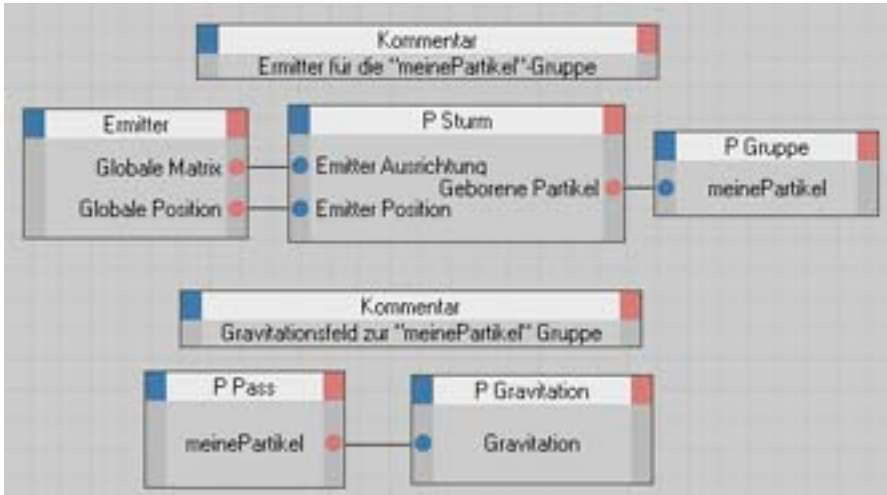
Nun müßt Ihr natürlich noch eine Verbindung zwischen dem „P Pass“-Node und dem „P Gravitation“-Node im XPresso-Editor herstellen, damit das Gravitationsfeld endgültig weiß, welche Partikel beeinflusst werden sollen.



So noch einen Schritt und Ihr habt es geschafft!

Schaut mal in die 3D Ansicht. Ihr seht hier ein weißes Quadrat mit einem Pfeil. Das stellt das Gravitationsfeld und die Richtung dar, in welche das Gravitationsfeld wirkt. Dreht dieses noch um -90 Grad und verschiebt es etwas nach links unten. Der Pfeil muss nach unten zeigen!

Nun drückt mal die Abspieltaste. Fertig. Die Partikel fallen nach unten. So schwer war es doch gar nicht oder ?



Noch ein Tip: Es ist für die Übersichtlichkeit besser, wenn Ihr sogenannte „Remarks“ im XPresso Editor verwendet. Das sind Felder, die mit Kommentaren bestückt werden können und somit die Übersichtlichkeit erhöht:

Re.Maustaste->Neuer Node->Allgemein->Allgemein->Kommentar


Hier die Datei mit der fertigen Schaltung: ***tp_tut04(fertig).c4d***

5 | Reflektoren

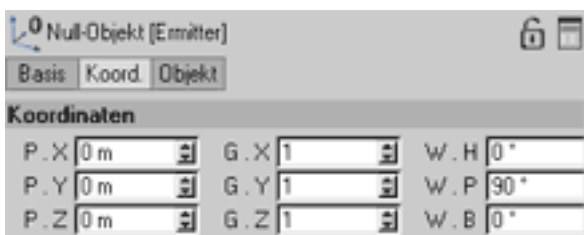
Das folgende Tutorial wird richtig schön kurz, da wir im Groben und Ganzen die fertige Szene aus dem Basis-Tutorial 4 übernehmen können. Bitte holt sie euch, falls Ihr sie nicht mehr auf der Festplatte habt : **tp_tut04(fertig).c4d**

Einen Reflektor/Deflektor setzt man dann ein, wenn die Partikel von anderen Objekten abprallen sollen. Reflektoren/Deflektoren für Partikel können beliebige Objekte einer Szene sein.

Wie binden wir nun einen Reflektor ein ?

Öffnet zunächst den XPresso Editor mittels Doppelklick auf das entsprechende Symbol im Objektmanager () und löscht dort den „P Gravitation“-Node.

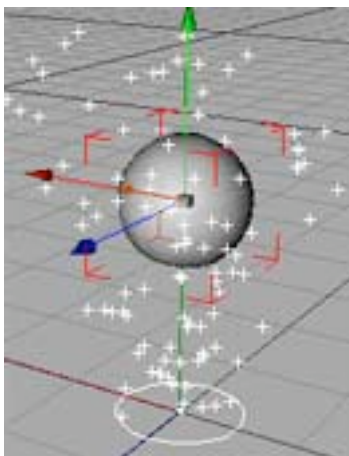
Im Objekt-Manager löscht Ihr noch das Null-Objekt, welches das Gravitationsobjekt darstellt und dreht das Nullobjekt für den „P Sturm“-Node so, dass es nach oben zeigt, also die Partikel nach oben abgegeben werden.



So, nun können wir endlich loslegen :

Zunächst brauchen wir ein Objekt, von dem die Partikel abprallen können. Wir nehmen hier eine Kugel mit einem Radius von 50m.

Positioniert die erstellte Kugel mit etwas Abstand über dem Emitter und drückt die Abspieltaste

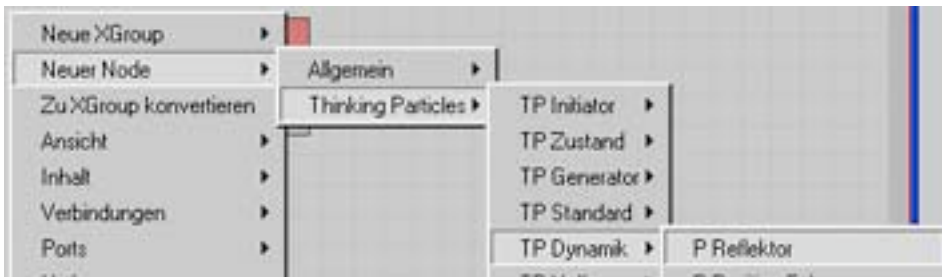


Die Partikel fliegen doch glatt durch! ...die bösen Dinger!

Damit man einen Reflektor (in unserem Fall die Kugel) in eine Partikelanimation einbinden kann, sieht „Thinking Particles“, wie sollte es auch anders sein, wieder einen Node vor. Den „P Reflektor“-Node.

Genau den werden wir jetzt im XPresso-Editor erstellen.

Neuer Node->Thinking Particles->TP Dynamik->P Reflektor



Dem neu erstellten "P Reflektor"-Node muß nun noch das Objekt zugewiesen werden, mit welchem die Partikel kollidieren sollen.

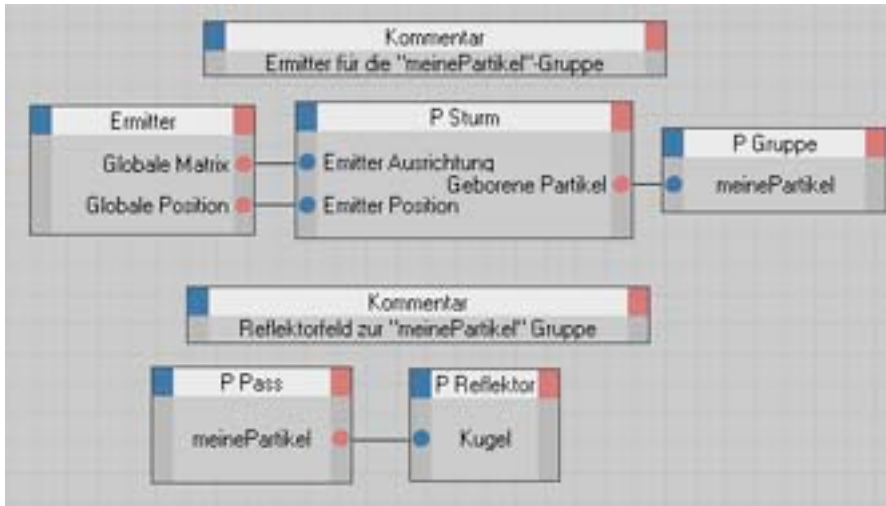
Dazu markiert Ihr den "P Reflektor"-Node im XPresso-Editor. Im Attribute-Manager seht Ihr nun ein Feld mit dem Namen "Objekt". Dreimal dürft Ihr raten, was da wohl rein gehört.

Zieht also das Kugel-Objekt aus dem Objekt-manager mit der gedrückten linken Maustaste in dieses Feld.



Damit das Objekt auch tatsächlich als Reflektor benutzt wird, müßt Ihr im Feld Reflektor Typ noch "Objekt" auswählen (siehe Bild oben).

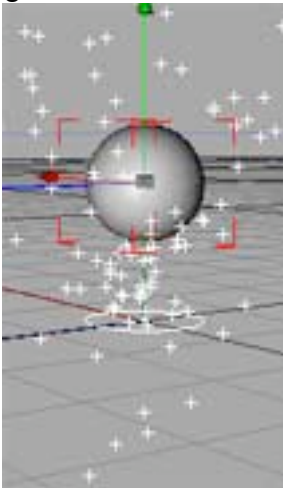
Jetzt müßt Ihr nur noch den Eingang des "P Reflektor"-Nodes mit dem Ausgang des "P Pass"-Nodes im XPresso-Editor verbinden, damit der "P Reflektor"-Node auch darüber informiert ist, welche Partikel er abweisen soll.



Fertig! Drückt mal auf die Abspieltaste!

Tja, war wohl nichts.... jetzt kommt der Pferdefuss der Geschichte. Als Reflektoren dienen nur Polygonobjekte!!!! Konvertiert daher die Kugel in ein Polygonobjekt.

Wenn Ihr jetzt auf die Abspieltaste drückt, prallen die entsprechenden Partikel von der Kugel ab!



Das wars mal wieder. Im nächsten Basis-Tutorial geht es darum, wie Ihr Partikel dazu bringt, einem Objekt zu folgen, bevor es ab Tutorial 7 etwas komplizierter wird. Also man sieht sich im Tutorial 6!


Hier die Datei mit der fertigen Schaltung: ***tp_tut05(fertig).c4d***

6 Partikel folgen Objekten

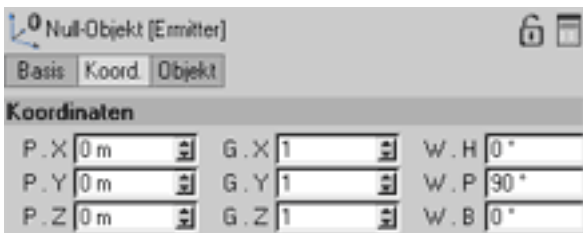
Auch dieses Tutorial ist relativ kurz, da wir wieder die fertige Szene des Basis-Tutorial 4 übernehmen können. Bitte holt sie euch hier , falls Ihr sie nicht mehr auf der Festplatte habt : ***tp_tut04(fertig).c4d***

In diesem Tutorial werden wir dafür sorgen, daß die Partikel auf ein bestimmtes Ziel zufliegen und dieses auch verfolgen, wenn das Zielobjekt in Bewegung ist.

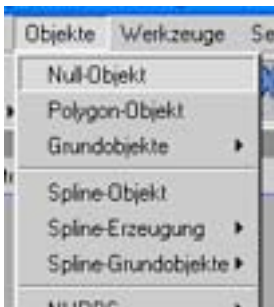
Zunächst müsst Ihr auch hier wieder die Gravitation aus der Szene entfernen :

Öffnet zunächst den XPresso Editor mittels Doppelklick auf das entsprechende Symbol im Objektmanager () und löscht dort den „P Gravitation“-Node.

Im Objekt-Manager löscht Ihr noch das Null-Objekt, welches das Gravitationsobjekt darstellt und dreht das Nullobjekt für den „P Sturm“-Node so, dass es nach oben zeigt, also die Partikel nach oben abgegeben werden.



Wir erstellen erstmal ein "Zielobjekt". Das kann ein beliebiges Objekt Eurer Wahl sein. Wir nehmen hier einfach ein Nullobjekt.



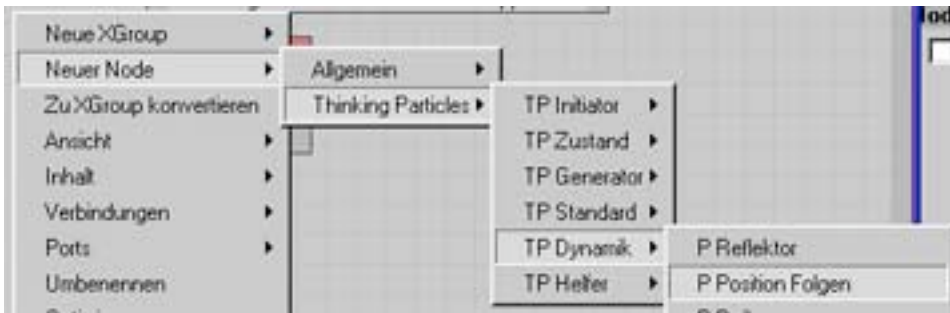
Positioniert das Ziel mit etwas Abstand zum Emitter in der Szene und gebt ihm den Namen "Flugziel".

Wenn Ihr die Abspieltaste drückt, dann ignorieren die Partikel das Ziel. Das werdet Ihr Euch ja auch schon gedacht haben...

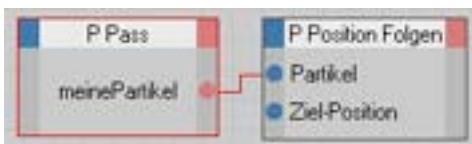
Auch hier bedarf es eines dafür vorgesehenen Nodes und zwar den "P Position Folgen"-Node.

Den müsst Ihr im XPresso-Editor erzeugen:

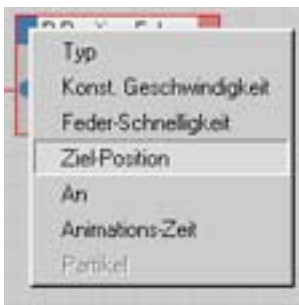
Neuer Node->Thinking Particles->TP Dynamik->P Position Folgen



Damit der Node weiß, auf welche Partikel-Gruppe er sich beziehen soll, verbindet Ihr den vorhandenen "Partikel"-Eingang mit dem "meinePartikel"-Ausgang des "P Pass"-Nodes.

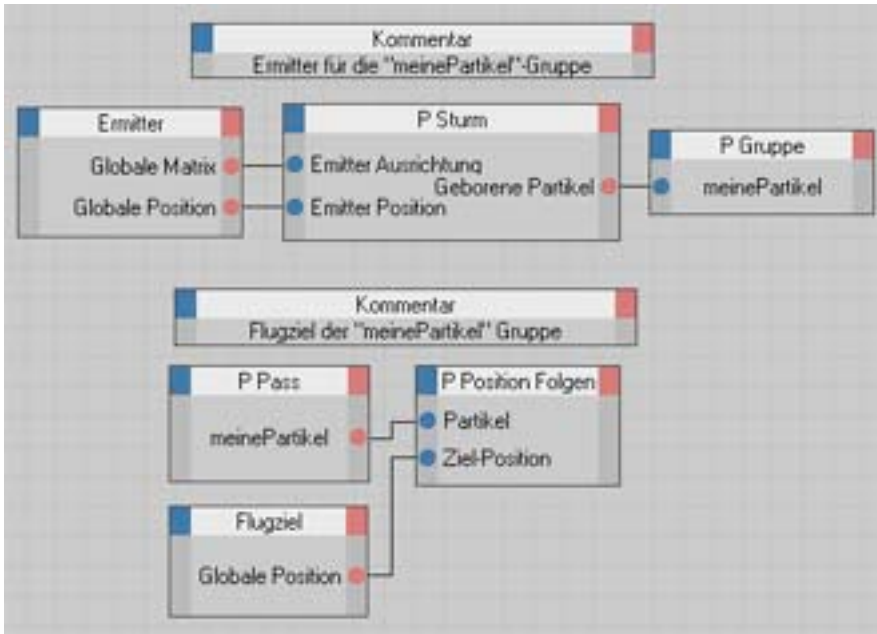


Natürlich müssen wir dem "P Partikel Folgen"-Node auch noch mitteilen, welches Ziel "anvisiert" werden soll. Dafür gibt es den Eingang "Ziel Position" im "P Partikel Folgen"-Node. Erstellt dieses mittels Klick in das blaue Eingangskästchen in der Titelzeile.



Nun fehlt nur noch das Zielobjekt in der XPresso-Schaltung. Zieht mit der gedrückten linken Maustaste das "Flugziel"-Objekt aus dem Objekt-Manager in den XPresso-Editor und erstellt im "Flugziel"-Node einen Ausgang mit dem Namen "Globale Position"(s. Bild unten). Der Ausgang stellt die Positionsdaten des Objektes zur Verfügung.

Verbindet noch den neu erstellten Ausgang mit dem "Ziel Position"-Eingang des "P Partikel Folgen"-Nodes, damit dieser Node informiert ist, welche Zielkoordinaten er an die Partikel weiterleiten soll.



Drückt auf die Abspieltaste und bewegt das Zielobjekt hin und her. Wunschgemäß visieren die Partikel das Ziel an! Leider sieht das doch etwas merkwürdig aus, da die Partikel das Ziel geradlinig ansteuern.

Der "Look" ist also etwas Verbesserungswürdig. Markiert im XPresso-Editor den "P Position Folgen"-Node. Im Attribute-Editor findet Ihr ein Feld mit dem Namen Typ. Stellt dort "Feder" ein und ändert den Wert "Feder-Schnelligkeit" auf "5".



Wenn Ihr nun wieder auf die Abspieltaste drückt und das Zielobjekt bewegt, dann sieht die Partikellaufbahn "organischer" und somit eher so aus, wie man es erwarten würde.

Na, war DAS kurz ?

Hier die Datei mit der fertigen Schaltung: ***tp_tut06(fertig).c4d***

7 | Objekte als Partikelemitter

Um Objekte als Partikelemitter verwenden zu können, muss man die Animation grundsätzlich anders anfangen. Eine Übernahme von Dateien aus den vorhergehenden Szenen ist daher nicht möglich.

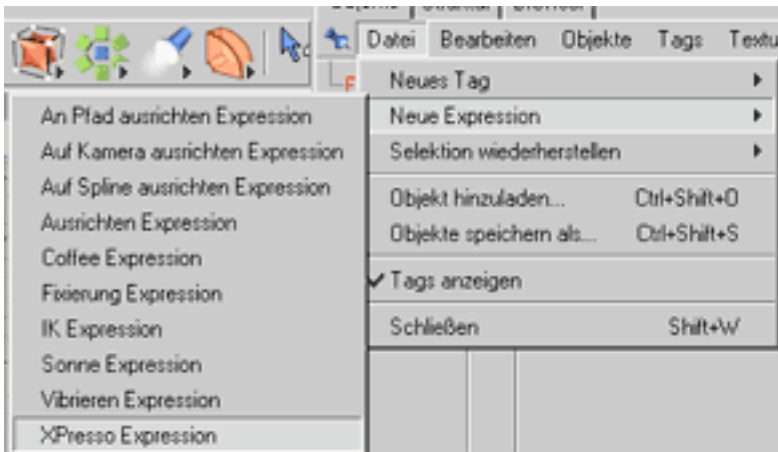
Es gibt zwei Möglichkeiten, Objekte als Partikel-Emitter zu missbrauchen. Man kann die Objekt-Oberfläche oder das Volumen eines Objektes verwenden. In diesem Tutorial werden wir uns zunächst um die Oberfläche kümmern.

Erstellt zunächst einmal eine neue Szene. Datei->Neu

Als Emitter verwenden wir eine Kugel mit einem Radius von 100m, geben ihr den Namen "Emitter" und wandeln sie in ein Polygonobjekt um.

Als nächstes müssen wir der Szene eine neue Expression hinzufügen.

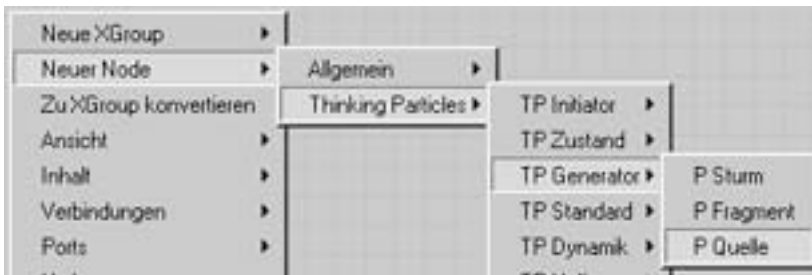
Objekt-Manager->Datei-Neue Expression->Presso Expression



Hinter dem Emitter-Objekt erscheint nun das mittlerweile altvertraute Zeichen ()

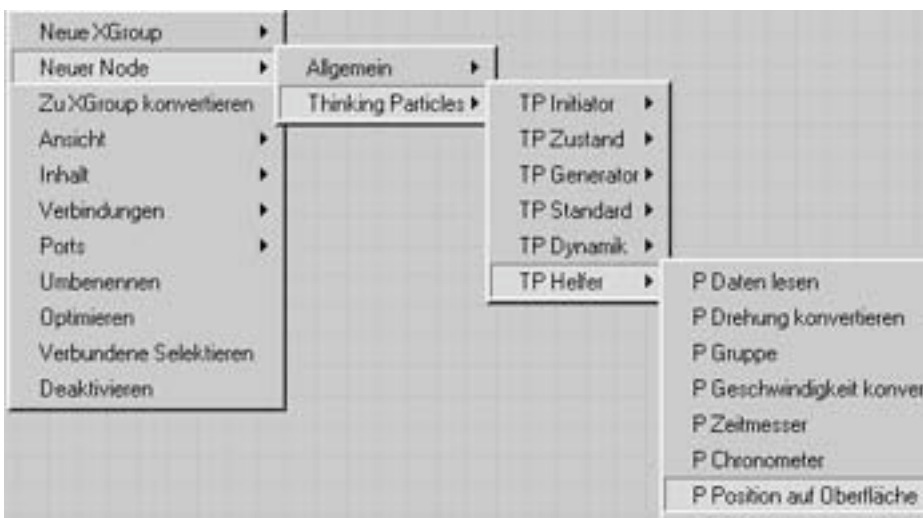
Öffnet nun den XPresso-Editor mittels Doppelklick auf das Symbol. Der leere Editor springt auf.

Als erstes müssen wir im XPresso-Editor wieder einen Emitter erstellen. Diesmal aber ausnahmsweise nicht den "P Sturm"-Node, sondern ein völlig neues. Erstellt ein "P Quelle"-Node!

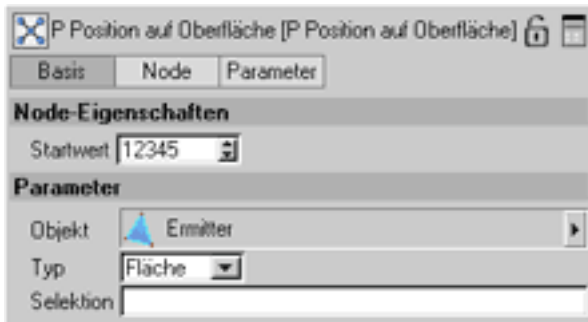
re.Maustaste->Thinking Particles->TP Generator->P Quelle

Bei dem "P Quelle"-Node handelt es sich sozusagen um den kleinen Bruder des "P Sturm"-Nodes. Er ist im Funktionsumfang eingeschränkter.

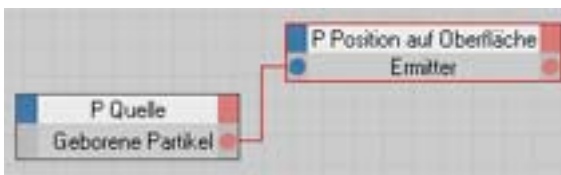
Um unser Emitter-Objekt der XPresso-Schaltung hinzuzufügen, benötigen wir ein weiteres Node. Den "P Position auf Oberfläche"-Node. Dieser liefert zufällig ausgewählte Positionen auf den Polygonen, Kanten oder Punkten eines Objektes (hier unsere Kugel). Grundsätzlich habt Ihr hier die Wahl zwischen den drei Varianten. Wir entscheiden uns für die Polygone. (Hinweis: Man kann die Partikel natürlich auch auf Selektionen beschränken!).

re.Maustaste->Thinking Particles->TP Helfer->P Position auf Oberfläche

Diesem Node müsst Ihr noch mitteilen, auf welches Objekt er sich beziehen soll. Zieht daher das "Emitter"-Objekt mit der gedrückten linken Maustaste aus dem Objekt-Manager in das "Objekt" Feld des "P Position auf Oberfläche" im Attribute-Manager...



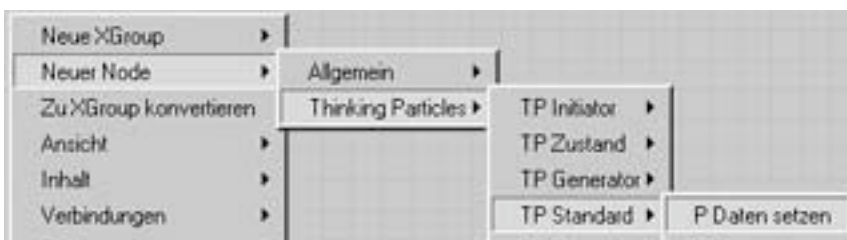
...und stellt eine Verbindung zwischen dem "Geborene Partikel" Ausgang des "P Quelle"-Nodes und dem "Emitter" Eingang des "P Position auf Oberfläche" her, damit auch der "P Quelle"-Node eine Positionsangabe erhält.



Leider reicht das nicht. Ein neuer Node muß her! Und zwar folgender : "P Daten Setzen". Warum ?

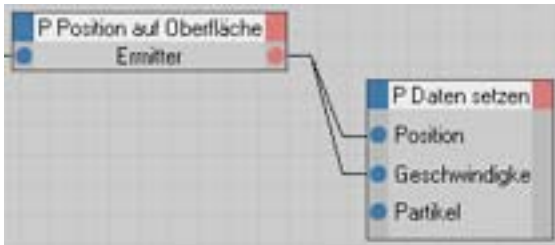
Mit diesem Node ist es möglich, einem Partikelstrom verschiedene Parameter zuzuweisen - selbst - nachdem sie den Emitter verlassen haben.

Neuer Node->Thinking Particles->TP Standard->P Daten Setzen



Damit wir die Partikel platzieren können, benötigen wir Informationen über Position und Geschwindigkeit der Partikel. Dafür erzeugen wir zwei neue Eingänge im "P Daten Setzen"-Node: Position und Geschwindigkeit.

Da der Geschwindigkeits-Vektor nicht nur die Geschwindigkeit sondern auch die Richtung der Bewegung festlegt, verbinden wir die neu erzeugten Eingänge des "P Daten Setzen"-Nodes mit dem "Emitter"-Ausgang des "P Position auf Oberfläche"-Nodes.



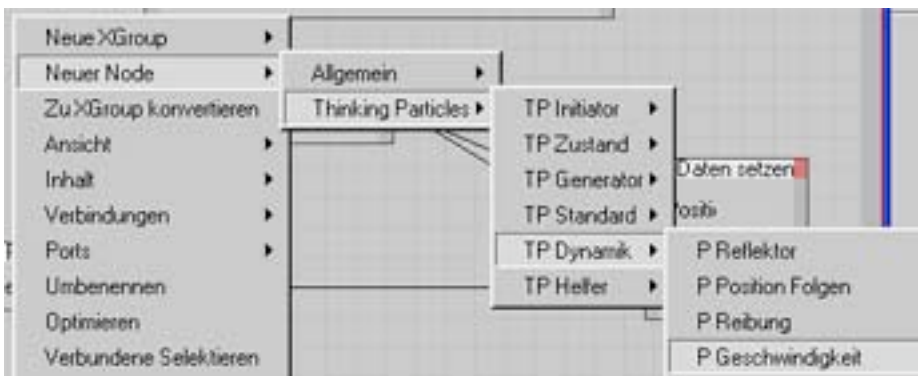
Damit der "P Daten Setzen"-Node weiß, welche Partikel er beeinflussen soll, muss zu guter Letzt noch eine Verbindung zum "P Quelle"-Node hergestellt werden. Zieht daher noch eine Linie vom "Geborene Partikel" Ausgang des "P Quelle"-Nodes zum "Partikel"-Eingang des "P Daten Setzen"-Nodes.

Fertig! Wenn Ihr nun die Abspieltaste drückt, dann seht Ihr, wie die Partikel von der Oberfläche ausgehen.

Mir Persönlich ist das aber etwas zu langsam. Nach einer Einstellungsmöglichkeit für die Geschwindigkeit suchen wir aber vergebens.

Deshalb muss schon wieder ein neuer Node her:

Neuer Node->Thinking Particles->TP Dynamik->P Geschwindigkeit



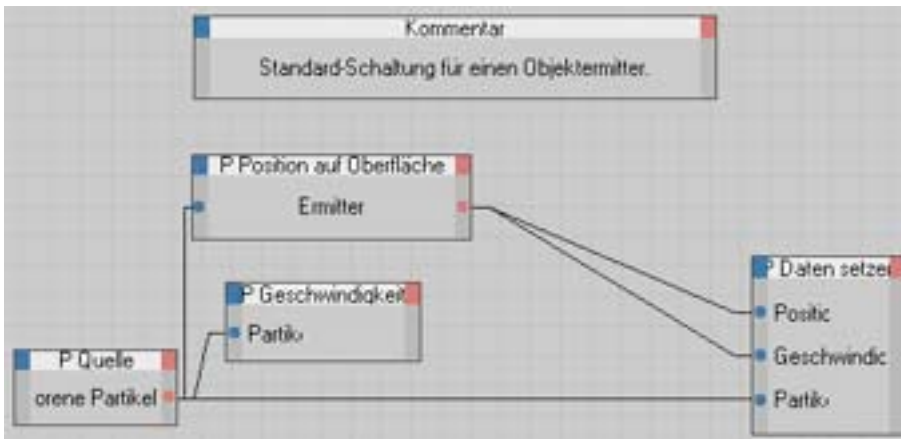
Mit diesem Node kann man, wie der Name schon sagt, die Geschwindigkeit von Partikeln beeinflussen.

Wenn Ihr noch den "Partikel" Eingang des "P Geschwindigkeit"-Nodes mit dem "Geborene Partikel" Ausgang von "P Quelle" verbindet, seid Ihr fast fertig.

Die Geschwindigkeit könnt Ihr nun im "Geschwindigkeit-Feld" des Attribute-Editors des "P Geschwindigkeit"-Nodes ändern (hier 500). Damit Ihr nicht die Richtung der Partikel beeinflusst, müßt ihr im Verändere-Feld noch "Geschwindigkeit" einstellen.



So, das war es nun wirklich. Unten seht Ihr noch mal die gesamte XPresso-Schaltung im Überblick:



Im nächsten und letzten Basis Tutorial Teil geht es um den Volumen-Emitter, bevor wir zu dem projektorientierten Tutorial übergehen.

8 | Objekte als Volumenemitter

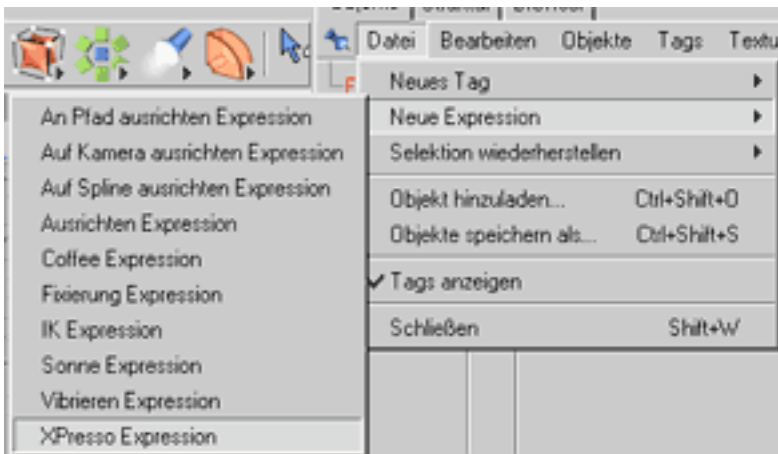
Einen Volumenemitter erstellt man auf ähnliche Weise, wie den Oberflächenemitter. Da die XPresso-Schaltung daher fast gleich aussieht, wird dieses Tutorial mal wieder etwas kürzer:

Erstellt zunächst einmal eine neue Szene. Datei->Neu

Als Emitter verwenden wir wieder eine Kugel mit einem Radius von 100m, geben ihr den Namen "Emitter" und wandeln sie in ein Polygonobjekt um.

Anschließend fügen für eine neue XPresso-Expression ein:

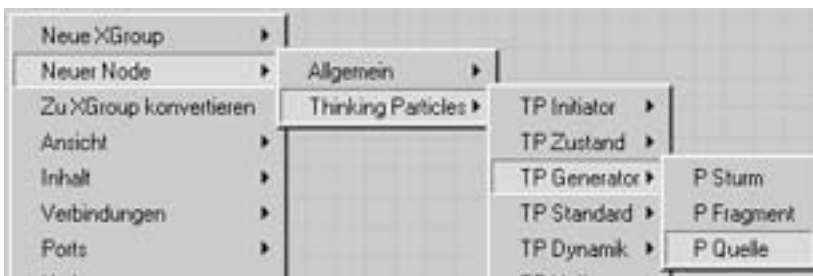
Objekt-Manager->Datei-Neue Expression->Presso Expression



Hinter dem Emitter-Objekt erscheint das mittlerweile altvertraute Zeichen () über den Ihr den XPresso-Editor öffnet.

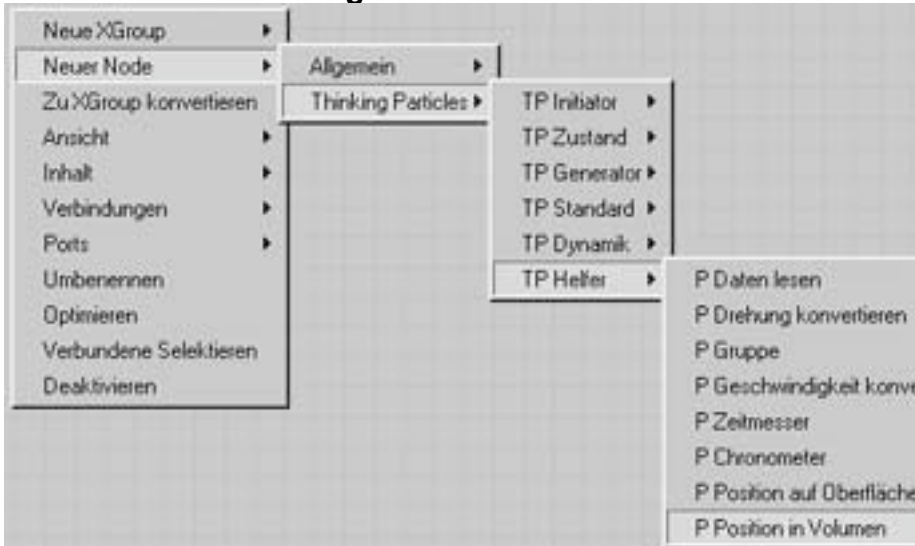
Als erstes müssen wir im XPresso-Editor einen Emitter erstellen. Wieder nehmen wir den "P Quelle"-Node!

re.Maustaste->Thinking Particles->TP Generator->P Quelle

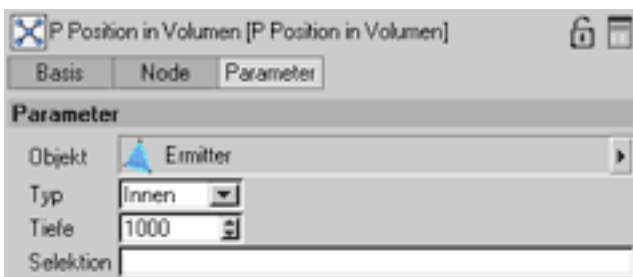


Um unser Emitter-Objekt der XPresso-Schaltung hinzuzufügen, benötigen wir ein weiteres Node. Diesmal den "P Position in Volumen"-Node. Dieser liefert zufällig ausgewählte Positionen im Volumen eines Objektes (hier unsere Kugel).

re.Maustaste->Thinking Particles->TP Helfer->P Position in Volumen



Diesem Node müsst Ihr noch mitteilen, auf welches Objekt er sich beziehen soll. Zieht daher das Emitter"-Objekt mit der gedrückten linken Maustaste aus dem Objekt-Manager in das "Objekt" Feld des "P Position in Volumen" im Attribute-Manager...

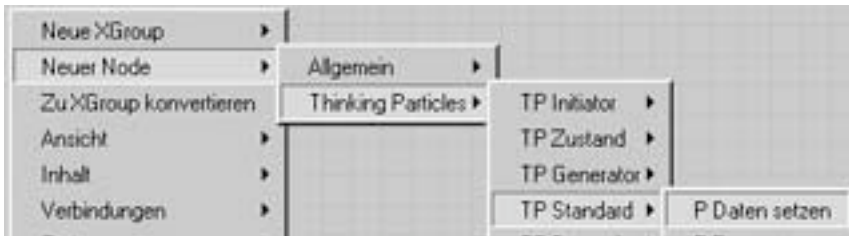


...und stellt eine Verbindung zwischen dem "Geborene Partikel" Ausgang des "P Quelle"-Nodes und dem "Emitter" Eingang des "P Position in Volumen" her, damit auch der "P Quelle"-Node eine Positionsangabe erhält.



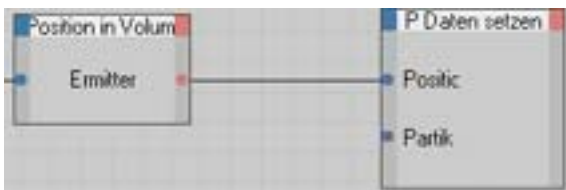
Leider reicht das nicht. Ein neuer Node muß her! Hier tut es wieder der "P Daten Setzen"-Node.

Neuer Node->Thinking Particles->TP Standard->P Daten Setzen

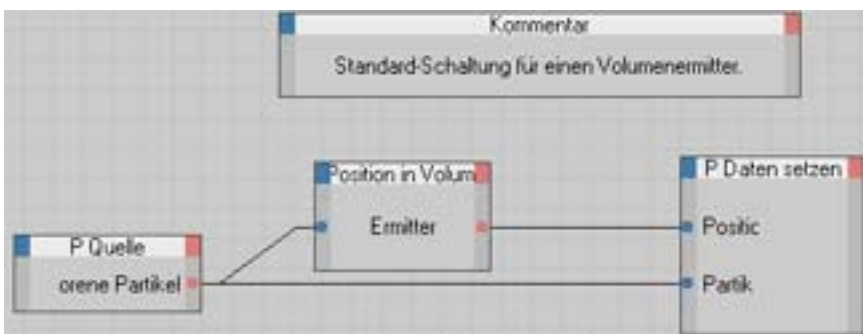


Damit wir die Partikel platzieren können, benötigen wir Informationen über die Position. Dafür erzeugen wir einen neuen Eingang im "P Daten Setzen"-Node und zwar "Position".

Anschließend verbinden wir den neu erzeugten Eingang des "P Daten Setzen"-Node mit dem "Emitter"-Ausgang des "P Position in Volumen"-Nodes.



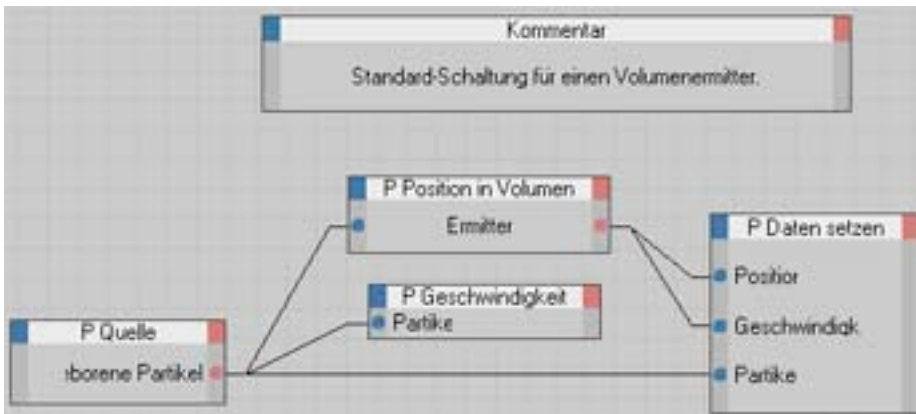
Damit der "P Daten Setzen"-Node weiß, welche Partikel er beeinflussen soll, muss auch hier noch eine Verbindung zum "P Quelle"-Node hergestellt werden. Zieht daher eine Linie vom "Geborene Partikel" Ausgang des "P Quelle"-Nodes zum "Partikel"-Eingang des "P Daten Setzen"-Nodes.



Fertig! Zur besseren Kontrolle solltet Ihr noch die Emitter-Kugel unsichtbar machen. Wenn Ihr nun die Abspieltaste drückt, dann seht Ihr, wie die Partikel innerhalb des Kugelvolumens entstehen.

Auch hier könnt Ihr den Partikeln eine Geschwindigkeit geben, indem Ihr ein "P Geschwindigkeit"-Node und einen Ausgang namens "Geschwindigkeit" beim "P Daten Setzen"-Node erstellt und diesen mit dem "P Position auf Oberfläche" Partikel Ausgang verbindet (siehe Tutorial 7).

Hier noch mal alles im Überblick:



So das war das letzte Basis Tutorial. Nun müßtet Ihr mit dem grundsätzlichen Steuern der Partikel im XPresso-Editor vertraut sein. Wenn Ihr wollt, könnt Ihr mit den folgenden Projekt-Tutorials weitermachen.

Hier die Datei mit der fertigen Schaltung: ***tp_tut08(fertig).c4d***

P1 | Projekt "Schneekugel"

Mit den nächsten Tutorials möchte ich Euch zeigen, wie man den Inhalt aus den Basis-Tutorials auch praktisch einsetzen kann.

Im ersten Projekt Tutorial werden wir eine Schneekugel bauen, die man in einem bestimmten Rahmen in "Realtime" schütteln kann. Um den "Schnee" in die Kugel zu bekommen, werden wir den Inhalt der Basis Tutorials 1-5 benötigen. Da ich dort schon alles ausführlich erklärt habe, werde ich mich hier etwas kürzer fassen und am Anfang eines Themenwechsels in Klammern auf das jeweilige Basis Tutorial hinweisen.

Damit Ihr wisst worauf Ihr Euch einlasst, zeige ich Euch erstmal einen Screenshot - alternativ habe ich auch gleich eine Animation beigefügt, damit Ihr auch die Partikel in Aktion sehen könnt:



Die Animation findet Ihr in bei den mitgelieferten Dateien!

Sicherlich kann man an der Partikelbewegung noch so einiges "drehen". Aber das würde den Rahmen dieses Tutorials sprengen. Experimentiert anschließend einfach mal mit den Nodes und deren Einstellungsmöglichkeiten rum.

Zunächst müsst Ihr die Start-Szene öffnen : ***tp-projekt01(start).zip***

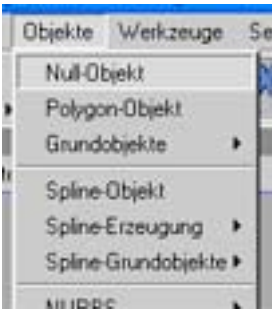
Wie Ihr seht, enthält die Szene bis auf das Partikelsystem alle Objekte, die Ihr für die Schneekugel braucht.

So, los geht's:

Damit wir überhaupt ersteinmal Partikel bekommen, brauchen wir einem Partikel-Emitter (Basis Tutorial 1).

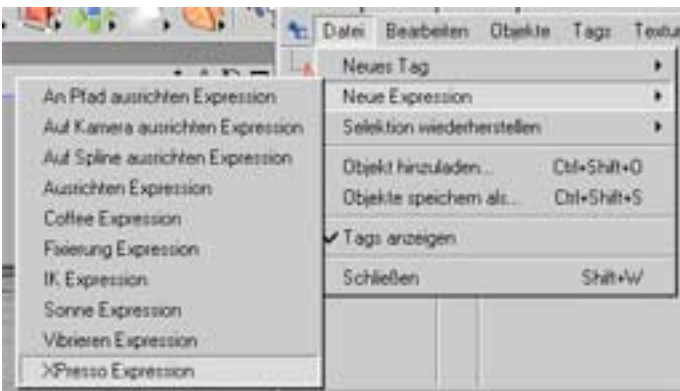
Erstellt dazu Null-Objekt.


Objekte-> Null-Objekt



Gebt dem Null-Objekt noch den Namen "Schnee (Emitter)" und fügt diesem eine neue Expression hinzu.

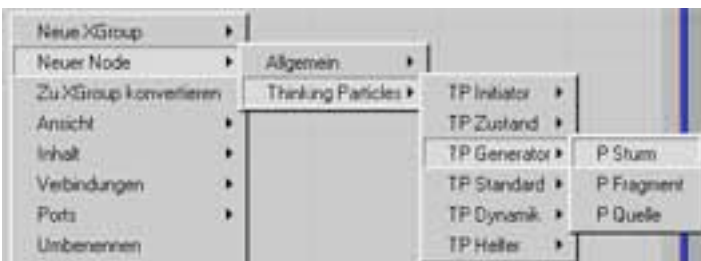
Objekt-Manager->Datei->Neue XPression->XPresso Expression



Neben dem "Schnee (Emitter)" erscheint das Symbol .

Im geöffneten XPresso-Editor erstellen wir einen PSturm-Node.

re. Maustaste->Neuer Node->Thinking Particles->TP Generator->PSturm



Zieht nun den zuvor erstellten "Schnee (Emitter)" in den XPresso-Editor und erstellt dort die Ausgänge Globale Matrix und Globale Position.

Weiterhin benötigen wir am P Sturm-Node noch den Eingang Emitter Ausrichtung.

Null-Objekt/Emitter :

Ausgang :

Globale Matrix

Koordinaten->Globale Position-> Globale Position

PSturm :

Eingang : Emitter Ausrichtung

Nun müsst Ihr noch die entsprechenden Ein- und Ausgänge verbinden.



So, nun weiss der Emitter, wo er ist und in welche Richtung er schaut(Wie nicht anders zu erwarten, schaut er natürlich in die Falsche!).

Damit Ihr besser erkennen könnt, wie der Emitter ausgerichtet ist, könnt Ihr die Abspieltaste drücken. Markiert den Emitter im Objekt-Manager - dreht und verschiebt diesen, bis er nach unten zeigt und sich im oberen Bereich der Schneekuppel befindet. Der Emitter sollte dabei nicht zu nah an der Kuppel liegen! Zur Orientierung habe ich meine Daten einmal abgebildet :

Koordinaten					
P . X	0 m	G . X	1	W . H	0°
P . Y	750 m	G . Y	1	W . P	-90°
P . Z	3 m	G . Z	1	W . B	0°

Mit dem Skalieren Werkzeug solltet Ihr dem Schnee (Emitter) noch eine ovale Form geben, damit von Anfang an eine gute Verteilung der Partikel gegeben ist.

Ihr werdet sicher alle schon einmal eine Schneekugel gesehen haben oder Ihr habt sogar eine im Regal stehen.

Es gibt bei einer Schneekugel keinen Emitter, der ständig neuen Schnee erzeugt. Das wollen wir hier auch nicht ändern. Daher werden wir dem Emitter mitteilen, dass er im Bild 1 alle Partikel erzeugen soll und dann in Rente gehen kann, weil er nicht mehr benötigt wird.

Dazu brauchen wir eine Überprüfung der aktuellen Bildnummer (Basis Tutorial 2).

Dafür erstellen wir die Nodes Zeit und Vergleich.

re. Maustaste->Neuer Node->Allgemein->Allgemein-> Zeit

re. Maustaste->Neuer Node->Allgemein->Logik->Vergleich

Im vorliegenden Fall ist eine Angabe der Zeit in Bildern erforderlich. Daher benötigen wir im Zeit-Node einen Ausgang, der die aktuelle Bildnummer ausgibt. Erstellt einen Bild Ausgang. (Die Ausgänge Real und Zeit könnt Ihr löschen)

Verbindet den neuen "Bild-Ausgang" des Zeit Nodes mit dem "Eingang 1" des Vergleichs-Nodes.

Markiert den Vergleich-Node. Im Attribut-Editor müsst Ihr noch kleiner/gleich (==) und für Eingang 2 den Betrag „1“ eintragen.

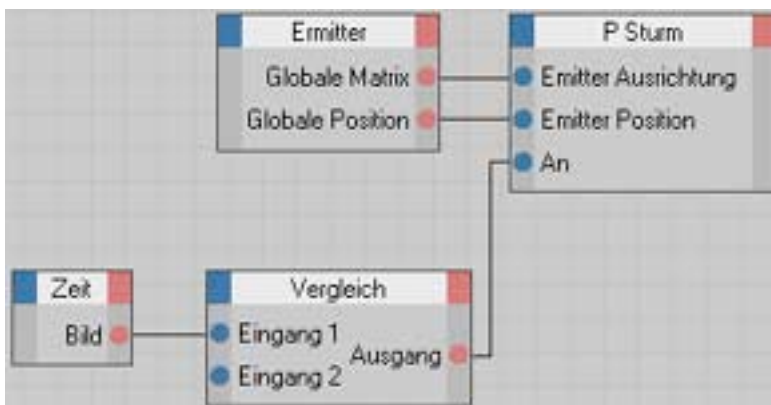
Der Vergleichs-Node prüft nun, ob die aktuelle Bildnummer gleich 1 ist.

Das alleine reicht natürlich nicht. wir müssen natürlich noch eine Verbindung zum Emitter herstellen, damit er auch weiss, ob es Zeit zum schlafen ist.

Erstellt im PSturm-Node daher einen „An“ Eingang.....

Linke Maustaste auf das blaues Feld->An

....und verbindet des Ausgang des "Vergleich"-Nodes mit dem "An-Eingang" des PSturm Nodes. So, der Emitter hat jetzt auch einen eigenen Wecker...



Damit eine vorher bestimmte Anzahl von Partikeln abgegeben wird, müsst Ihr noch (bei markiertem PSturm Node) den Modus "Shot" im Attribute-Editor einstellen und den Shot-Wert auf 300 setzen. Für die Erstellungsphase ist das erstmal genug. Beim Rendern habe ich die Partikelzahl auf 5000 einstellen müssen. Das könnt Ihr aber später nachholen.

Damit die Partikel auch genug Zeit haben sich in der Kuppel zu bewegen, solltet Ihr schon jetzt die Lebenszeit der Partikel auf 1000 stellen. In diesem Rahmen müsst Ihr auch die maximale Bilderzahl in den Dokument-Voreinstellungen auf den gleichen Wert setzen.

Da wir nun schon einmal in diesen Einstellungen sind, werden wir den Partikeln gleich eine Rotation verpassen, damit es etwas ungleichmäßiger wird. Bitte setzt daher noch den Wert für die Rotation auf 30 und die Rotations-Variation auf 40%

Der Effekt wird aber erst dann sichtbar, wenn wir auch die Schneeflocken verwenden.

Wenn Ihr wollt, dürft Ihr nun mal auf die Abspieltaste drücken. Wenn alles geklappt hat, dann müsstet am Anfang der Animation einmal 300 Partikel nach "unten" abgestossen werden.

Nachdem wir unseren Emitter nun soweit haben, dass genau 300 Partikel im Bild 1 erstellt werden, müssen wir uns darum kümmern, dass die Partikel auch in der Kuppel bleiben (Basis Tutorial 5).

Wenn Ihr im Objekt Manager in das Schneekugel Objekt hinein schaut, werdet Ihr ein Objekt namens Kuppel (Reflektor) finden.

Dieses Objekt werden wir dazu verwenden, um die Partikel zu zähmen.

Um auf die Partikel zugreifen zu können, benötigen wir einen P Pass-Node.

Re.Maustaste->Neuer Node->Thinking Particles->TP Initiator->P Pass

Weiterhin brauchen wir den P Reflektor-Node, um die Partikel abprallen zu lassen.

Neuer Node->Thinking Particles->TP Dynamik->P Reflektor

Jetzt müssen wir dem P Reflektor-Node noch mitteilen, welches Objekt er als Reflektor verwenden soll.

Dazu markiert Ihr den "P Reflektor"-Node im XPresso-Editor. Im Attribute-Manager seht Ihr ein Feld mit dem Namen "Objekt".

Zieht das Kuppel (Reflektor)-Objekt aus dem Objekt-manager in dieses Feld.

Damit das Objekt auch tatsächlich als Reflektor benutzt wird, müßt Ihr im Feld Kollisionstyp noch "Objekt" auswählen.

Jetzt müßt Ihr nur noch den Eingang des "P Reflektor"-Nodes mit dem Ausgang des "P Pass"-Nodes im XPresso-Editor verbinden, damit der "P Reflektor"-Node auch darüber informiert ist, welche Partikel er abweisen soll.

Nun werden wahrscheinlich Fragen auftreten, warum ich nicht die eigentliche Kuppel benutzt habe.

Genau das habe ich Stundenlang vergeblich versucht!

Später erfuhr ich, "...dass es je nach Winkel und Einstellung des oberflächen Wertes, aufgrund von Rechenungenauigkeiten passieren kann, dass Partikel durch die Oberfläche "flutschen". Das ist eine Einschränkung sämtlicher Kollisionserkennungsverfahren die insbesondere dann zum tragen kommt wenn Partikel längere Zeit parallel an einer Oberfläche entlanggleiten...." (ein fast Zitat).

Daher habe ich zusätzlich zwei nahezu gleich große Kuppeln erstellt, die innere etwas stärker unterteilt. Durch die 2 Kollisionsflächen, die sehr nah bei einander liegen, ist das Risiko, das Partikel durch die Oberfläche gehen, minimiert... aber nicht ausgeschlossen. Gehäuft tritt das problem auf, wenn die Schneekugel rotiert und die Partikel am Glas entlanggleiten (Bitte noch nicht testen!).

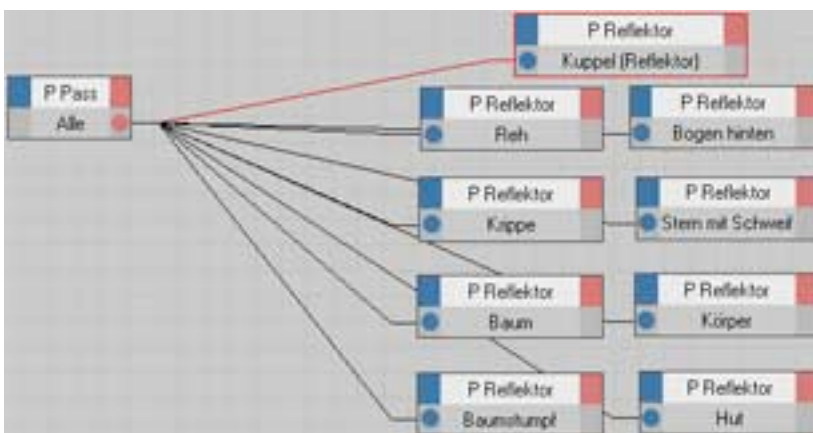
Das ist sicher nicht die Königsvariante aber für dieses Tutorial optimal und ausreichend!

Jetzt kommt eine Fleißarbeit :

Erstellt im XPresso-Editor acht (8) weitere "P Reflektor-Nodes" und verbindet diese mit dem Ausgang des P Pass-Nodes.

Anschließend weist ihr jedem der folgenden Polygonobjekte des Schneekuppel-Objektes jeweils einem der acht "P Reflektor-Nodes" zu und setzt den Kollisionstyp auf Objekt:

Reh, Krippe, Baum, Baumstumpf, Bogen hinten, Stern mir Schweif, Körper (im Schneemann), Hut (im Schneemann)



Wenn Ihr nun auf die Abspieltaste drückt, müssten die Partikel von allen Oberflächen inkl. der Kuppel abprallen.

Tip:

Die acht Reflektor-Objekte könnt Ihr schneller erstellen:

Erstellt ein P Reflektor Node, setzt den Kollisionstyp und kopiert diesen noch 7 mal in dem Ihr den Node mit der gedrückten Strg Taste an eine andere Stelle zieht.

Noch drei Nodes und Ihr seid erledigt :-)

Beim Ansehen der Partikel-Kollisionen wird Euch aufgefallen sein, dass die Partikel an sich keine Anstalten machen, sich endlich mal zum Boden zu begeben...

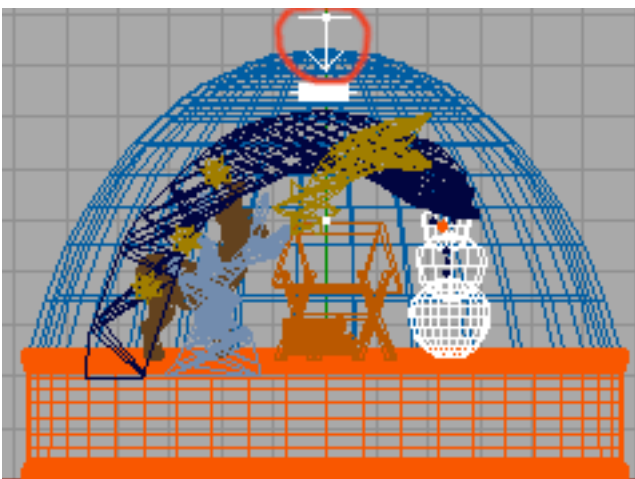
Nun ist daher an der Zeit, ihnen Manieren beizubringen. Ein P Gravitation-Node muss her!(Basis Tutorial 4)

Auch der „P Gravitations“-Node muß an ein Objekt gekoppelt sein. Wie immer empfiehlt sich hier wieder ein Null-Objekt. Bitte nennt dieses „Schnee (Gravitation)“.

Re.Maust.->Neuer Node->Thinking Particles->TP Dynamik->P Gravitation

Anschließend müsst Ihr das "Schnee (Gravitation)"-Objekt noch dem "P Gravitation"-Node zuweisen, um eine Verbindung herzustellen. Zieht daher das "Schnee (Gravitation)"-Objekt vom Objektinspektor in das Objekt-Feld von „P Gravitation“ im Attribute-Manager.

Dreht das Gravitation-Objekt noch um -90 Grad und verschiebt es mittig über die Schneekugel. Der Pfeil muß nach unten zeigen! Nun müsst Ihr natürlich noch eine Verbindung zwischen dem „P Pass“-Node und dem „P Gravitation“-Node im XPresso-Editor herstellen.



...und weil es grade so schön ist, ein weiteres Node :

Damit dir Partikel nicht alle gleichmäßig linear umherfliegen, verwenden wir hier den P Blubbern-Node. Dieser Node versieht die Partikelbewegung mit einer (Sinus-) Schwingung. Dadurch erreicht man eine etwas "flutterhafte" Flugbahn.

Re.Maust.->Neuer Node->Thinking Particles->TP Dynamik->P Blubbern

Auch hier müsst Ihr den Eingang des P Blubbern-Nodes mit dem Ausgang des P Pass Nodes verbinden.

Als Amplitude habe ich 2 und als Geschwindigkeitswert 10 benutzt. Das könnt Ihr aber individuell nach Gefühl einstellen.

Wenn Ihr nun auf Abspielen drückt, fliegen die Partikel schon nicht mehr ganz so gleichmäßig.

Was jetzt noch fehlt, ist das Partikelobjekt selbst - der Schnee(Basis Tutorial 3)!

Ich habe mich im vorliegenden Fall für einen Würfel als Partikelobjekt entschieden, da die Form bei der richtigen Partikelgröße nicht genau erkennbar ist und es ja sogar etwas unnatürlich aussehen soll(mal ganz abgesehen von der geringen Polygonzahl).

Erstellt also einen Würfel mit den Massen $5m \cdot 5m \cdot 5m$ ohne Rundung und wandelt diesen in ein Polygonobjekt um und gebt ihm den namen "Schneeflocke".

Danach müssen wir ein P Porm Node erstellen, um den Partikeln, wie der Name schon sagt, eine Form zu geben.

Re.Maustaste->Neuer Node->Thinking Particles-TP Standard->P Form

Übergebt nun die erstellte Schneeflocke an das P Form-Node, indem Ihr die Schneeflocke mit gedrückter linker Maustaste aus dem Objekt-Magager in das Objekt-Feld des P Form-Nodes im Attribute-Manager zieht und auch hier als Umkugelradius den Wert „5“ einträgt. Die Schneeflocke solltet Ihr zum Rendern auf unsichtbar schalten.

Damit Ihr den P Form Node an den P Sturm Node anschließen könnt, müsst Ihr am P Sturm Node noch einen "Geborene Partikel"-Ausgang erzeugen und diesen mit dem P Form Node verbinden.

Als nächstes muss noch ein Partikel-Geometrie Objekt her. Ihr findet es hier :

Plug_ins->Thinking Particles->Partikel-Geometrie

In das Feld Partikel Gruppe zieht Ihr noch die Partikel Gruppe "Alle" aus dem Einstellungs-menü von Thinking Particles in das "Particle Gruppe" Feld des Partikel-Geometrie-Objektes (zur Erinnerung : XPresso-Editor-Zusätze->Thinking Particles->Einstellungen)

Damit die Schneeflocke auch die richtige Größe behält, solltet Ihr ebenfalls noch den Abmessungswert für die Partikel im P Sturm Node auf 5 setzen.

Nun noch das Partikel-Geometrieobjekt in "Schneeflocke (Geometrie)" umbenennen und die "Schneeflocke" dem "Schneeflocke (Geometrie)"-Objekt im Objekt Manager unterordnen (und auf unsichtbar beim Rendern Schalten).

Um der Geschwindigkeit und der Masse der jeweiligen Partikel gerecht zu werden, habe ich noch im P Reflektor-Node der Kuppel einen Wert von 80% bei Energie eingetragen. Die Energiewerte der anderen Reflektoren habe ich so gelassen, wie sie sind, um etwas Unruhe in die Szene zu bringen.

Das war das erste Projekt-Tutorial, um die Anwendungsmöglichkeiten der Basis-Tutorials aufzuzeigen.

Im Objektinspektor solltet Ihr noch den Partikelemitter der Schneekugel unterordnen, damit sich dieser mit der Kuppel bewegt. Das hat den Vorteil, dass die Partikel nicht in den leeren Raum gehen, wenn die Schneekugel bei Ablauf der eingestellten 1000 Bilder nicht an der richtigen Stelle ist.

Ich hoffe, es hat Euch ein wenig geholfen, das Prinzip von "Thinking Particles" leichter zu verstehen.

Sicherlich kann man das Partikelverhalten noch perfektionieren in dem man z.B. die Bewegung der Schneekugel auf die Partikel überträgt, ohne eine Kollision mit den Objekten (denn bei einer echten Schneekugel ist das aufgrund der Wasserfüllung so!).

Spielt vor allem mit den Werten der Nodes rum, um mehr aus den Partikeln heraus zu holen. Beachtet auch, dass es sein kann, dass ggf. ein paar Partikel aus der Kuppel rutschen. Diese kann man z.B. mit einem Vernichter "weg" machen oder Ihr variiert leicht den Animationspfad.

Für diejenigen, die keine Lust haben, das Tutorial durchzuarbeiten, wird es wahrscheinlich bald ein entsprechendes Video-Tutorial geben.

Hier gibt es die fertige Szene : ***tp_projekt01(fertig).zip***

Damit es etwas übersichtlicher wird, die XPresso-Schaltung "am Stück":

