



CINEMA 4D

RELEASE 11



MAXON

3D FOR THE REAL WORLD

Quickstart Manual

CINEMA 4D Release 11

Quickstart Manual

Programming	Christian Losch, Philip Losch, Richard Kurz, Aleksander Stoppel, Tilo Kühn, Per-Anders Edwards, Sven Behne, Wilfried Behne, Thomas Kunert, David O'Reilly, Paul Everett, Cathleen Bastian, Ole Kniemeyer, Kent Barber.
Plugin programming	Michael Breitzke, Kiril Dinev, David Farmer, Jamie Halmick, Reinhard Hintzenstern, Jan Eric Hoffmann, Eduardo Olivares, Nina Ivanova, Markus Jakubietz, Eric Sommerlade, Hendrik Steffen, Jens Uhlig, Michael Zeier, Matthias Bober, Markus Spranger, Michael Kloß, Ralph Reichl, Eberhard Michaelis.
Product management	Marco Tillmann, Bernd Lutz
Quickstart writers	Glenn Frey, Sven Hauth, Dirk Beichert, Fabian Rosenkranz
Layout	Oliver Becker, Scot Wardlaw, Oliver Krawczyk
Translation	Scot Wardlaw, Luke Stacy

Copyright © 1989 - 2009 by MAXON Computer GmbH All rights reserved.

This manual and the accompanying software are copyright protected. No part of this document may be translated, reproduced, stored in a retrieval system or transmitted in any form or by any means, electronic or mechanical, for any purpose, without the express written permission of MAXON Computer.

Although every precaution has been taken in the preparation of the program and this manual, MAXON Computer assumes no responsibility for errors or omissions. Neither is any liability assumed for damages resulting from the use of the program or from the information contained in this manual.

This manual, as well as the software described in it, is furnished under license and may be used or copied only in accordance with the terms of such license. The content of this manual is furnished for informational use only, is subject to change without notice, and should not be construed as a commitment by MAXON Computer. MAXON Computer assumes no responsibility or liability for any errors or inaccuracies that may appear in this book.

MAXON Computer, the MAXON logo, Sketch and Toon, CINEMA 4D, Hyper NURBS, C.O.F.F.E.E. are trademarks of MAXON Computer GmbH or MAXON Computer Inc. Acrobat, the Acrobat logo, PostScript, Acrobat Reader, Photoshop and Illustrator are trademarks of Adobe Systems Incorporated registered in the U.S. and other countries. Apple, AppleScript, AppleTalk, ColorSync, Mac OS, QuickTime, Macintosh and TrueType are trademarks of Apple Computer, Inc. registered in the U.S. and other countries. QuickTime and the QuickTime logo are trademarks used under license. Microsoft, Windows Vista, Windows 2003 Server and Windows NT are either registered trademarks or trademarks of Microsoft Corporation in the U.S. and/or other countries. LightWave 3D is a registered trademark of NewTek. 3D studio max and 3ds max are registered trademarks of Autodesk/Discreet Inc. UNIX is a registered trademark only licensed to X/Open Company Ltd. All other brand and product names mentioned in this manual are trademarks or registered trademarks of their respective companies, and are hereby acknowledged.

Contents

Welcome to CINEMA 4D	2
1. Introduction	2
2. General Information / Interface	3
3. Sample Images	13
4. Quick Tutorial – Arranging Objects	15
5. Quick Tutorial – Modeling	23
6. Quick Tutorial – Materials	30
7. Quick Tutorial – Lighting	40
8. Quick Tutorial – Animation	46
9. Quick Tutorial – Rendering	55
10. Quick Tutorial – Multi-Pass Rendering	62
11. Quick Tutorial – XPresso	69
Welcome to BodyPaint 3D	76
1. Introduction	76
2. General Information / Interface	77
3. Sample Images	79
4. Quick Tutorial – The Paint Wizard	80
5. Quick Tutorial – First Painting Lesson	82
6. Tips & Tricks	86
Welcome to Advanced Render (Optional module)	88
1. Introduction	89
2. General Information / Interface	89
3. Sample Images	92
4. Quick Tutorial – Global Illumination	94
5. Quick Tutorial – Caustics	97
6. Quick Tutorial – Depth of Field	100
7. Tips & Tricks	101
Welcome to PyroCluster	103
1. Introduction	103
2. General Information / Interface	103
3. Sample Image	105
4. Quick Tutorial – 10 Steps To Glory	106
5. Quick Tutorial – Optimize and Animate	109
6. Tips & Tricks	111
7. In Closing	112
Welcome to Sketch and Toon (Optional module)	114
1. Introduction	114
2. General Information / Interface	114
3. Sample Images	117
4. Quick Tutorial – Outlines	119

5. Quick Tutorial – Shaders and Tags	123
6. Tips & Tricks	127
Welcome To NET Render (Optional module)	128
1. Introduction	129
2. General Information / Interface	129
3. Sample Images	129
4. Quick Tutorial – Installation / Interface	130
5. Quick Tutorial – Jobs and Administration	132
Welcome to Dynamics (Optional module)	135
1. Introduction	135
2. General Information / Interface	135
3. Sample Images	136
4. Quick Tutorial – Rigid Bodies	137
5. Quick Tutorial – Soft Bodies	140
6. Tips & Tricks	141
Welcome To Thinking Particles (Optional module)	143
1. Introduction	143
2. General Information / Interface	144
3. Quick Tutorial – Particle Snow	147
4. Quick Tutorial – Objects as Particles	150
5. Tips & Tricks	155
6. In Closing	155
Welcome To MOCCA (Optional module)	157
1. Introduction	157
2. General Information/Interface (Joints and Rigging)	158
3. Quick Tutorial – Forward Kinematics / Inverse Kinematics (FK/IK)	163
4. Quick Tutorial – Visual Selector	166
5. Quick Tutorial – PoseMixer	169
6. Quick Tutorial – Morph Tool	173
7. Quick Tutorial – Muscles	175
8. Quick Tutorial – VAMP	178
9. Quick Tutorial – Dressing	183
10. Quick Tutorial – FBX Import / Export	187
Welcome to HAIR (Optional module)	191
1. Introduction	191
2. General Information / Interface	192
3. Sample Images	193
4. Quick Tutorial – Fur	194
5. Tips & Tricks	205
Welcome To MoGraph (Optional module)	206

- 1. Introduction 207
- 2. General Information / Interface 207
- 3. Sample Images 210
- 4. Quickstart Tutorial 210
- 5. Tips & Tricks 215
- Welcome To 3D Ghosting 217**
 - 1. Introduction 217
- Welcome to Projection Man 220**
 - 1. Introduction 220
- Welcome to Non-Linear Animation in the Timeline 224**
 - 1. Motion Layer 225
 - 2. Animation Layer 229

Note:

As a result of continued product development, differences between the current and printed documentation with regard to referenced files can occur. The most current versions can be found on the product DVD included in your order, or can be downloaded from the MAXON website or via the Online Updater.



Welcome to CINEMA 4D

After you have worked through this tutorial you will have a good basic user knowledge which you can apply to future projects.



© Dave Brinda - www.brinda.com

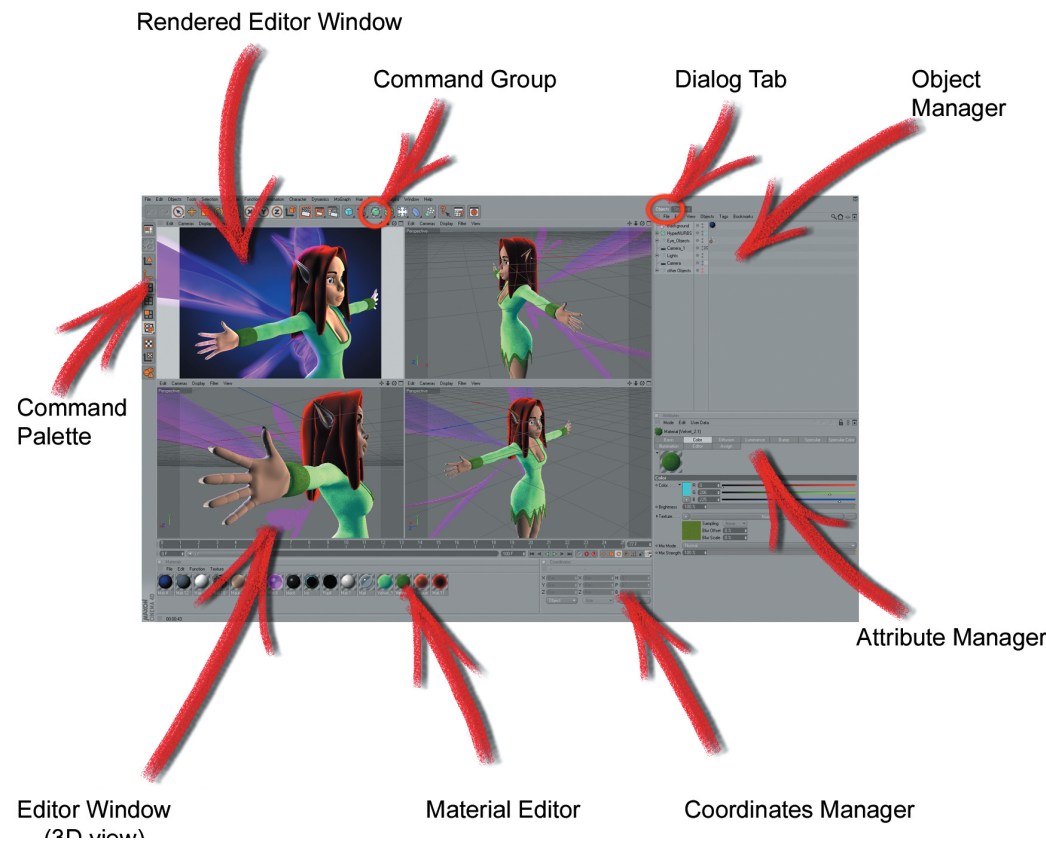
1. Introduction

No matter if you're just checking CINEMA 4D out or if you already own your own copy of CINEMA 4D or one of its complete XL or Studio Bundle packages, you already know about the incredible things CINEMA 4D can do. We have been working very closely with our customers for several years now in order to satisfy their needs and wishes. This has lead to the creation and introduction of new functionality, according to their needs. These ideas and concepts are then creatively implemented to satisfy the needs of our customers and those of the 3D markets. No matter if you work in the field of print, advertising, design, visualization or film, CINEMA 4D gives you all the tools you need to make your ideas reality. The intuitive interface and the ease with which CINEMA 4D can be learned makes entering the versatile world for 3D a snap. Whether you need character animation (MOCCA module) or a cartoony look of your renderings (Sketch and Toon module) – CINEMA 4D's modular setup lets you customize it to suit your needs. CINEMA 4D places a link between your job or hobby, and your creativity in the palm of your hand. You can create what your fantasy demands. CINEMA 4D will be your dependable partner.

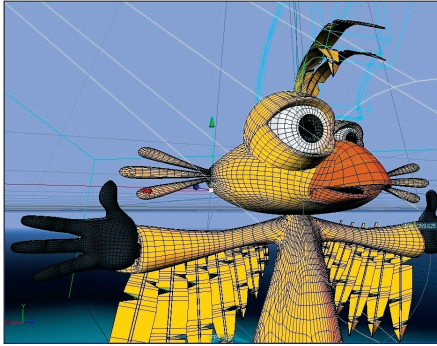
2. General Information / Interface

CINEMA 4D Release 11 offers many new functions that will again speed up and improve your workflow.

Let's start with the most important step - starting CINEMA 4D. After starting CINEMA 4D you will see an image similar to the following screenshot:



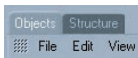
CINEMA 4D is divided into different working areas as follows: (starting at the top left clockwise)



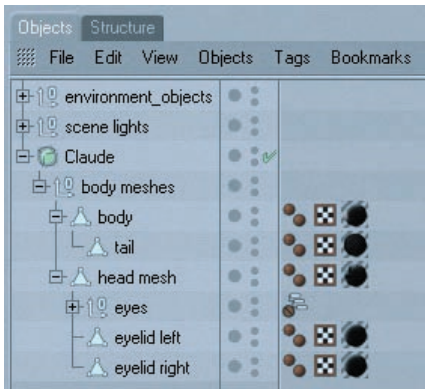
The Editor Window shows all objects contained in the scene, for example polygon objects, cameras, lights and bones and other deformers. You can render any view at any time to check your work.



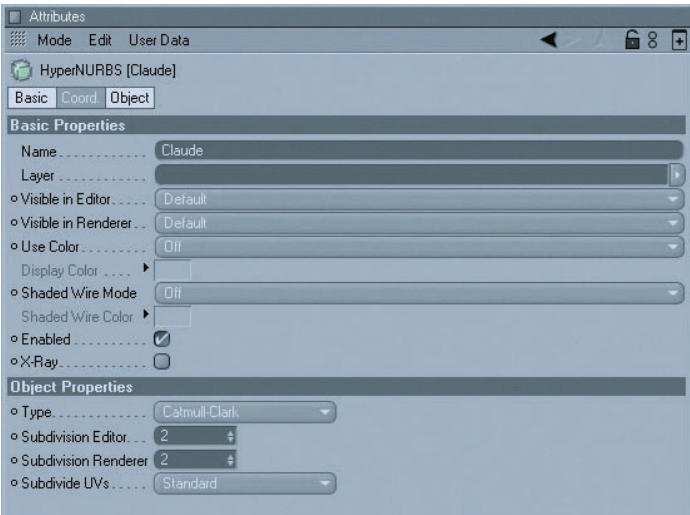
A **Group Icon** contains several attributes for one group which can be accessed by clicking with the left mouse button on the main icon. The group icon differs from normal icons in that you will see a small arrow in the lower-right corner.



A **Tab** indicates different windows or managers which are layered over each other. In each window or manager you will find different settings or attributes.



The **Object Manager** contains all of the scene’s objects. You use the Object Manager to set up a hierarchy, assemble objects, set tags for objects (small icons to the right of the Object Manager let you assign an object certain attributes), or to name objects. Included are polygon objects, lights, cameras, bones, deformers, splines and null objects (objects without geometry).



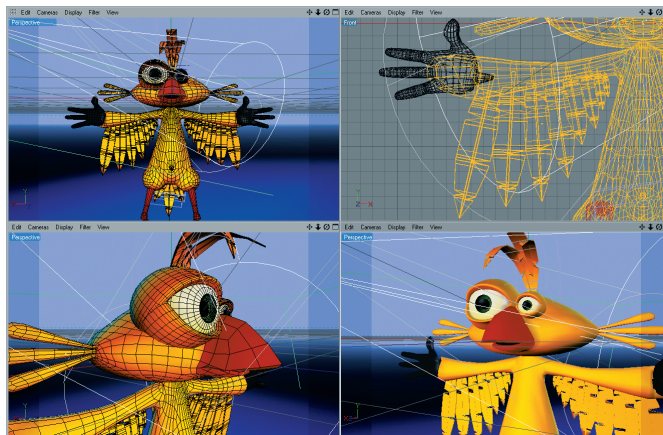
The Attribute Manager manages the attributes of each object or tool. This is where you can change the strength of the HyperNURBS subdivision (more about that later) or an object’s visibility in the editor window. The object’s coordinates can be found here as well as the tool setting such as the radius of the live selection and the “Only Select Visible Elements” option.



The **Coordinates Manager** lets you place, rotate or scale your objects. Enter the values in the given fields and confirm your entry with the “apply” button or simply press the return key.



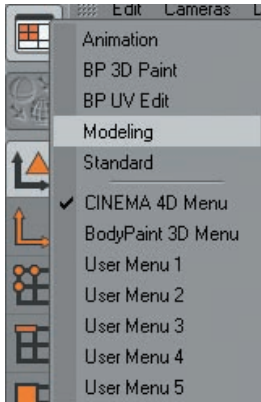
The Material Manager is used to display and manage all of a scene's materials and 3D shaders. Textures and parameters can be modified in the Attribute Manager or the Material Editor. Double-click the material to open the Material Editor to make changes to its individual material channels. Illumination strength, type of specular, strength of bump and more can be adjusted here as well. We'll cover this in detail in a later chapter.



By default, CINEMA 4D starts with four open Viewports. You can, however, add any number of Viewports to your layout. You can view your scene in different modes ranging from gouraud shading (includes any lights you have placed into the scene) to quick shading (displays your scene using only a default light, not lights you have placed into the scene), wire frame and more. This lets you adapt your editor window layout to your needs or your computer's processor speed.



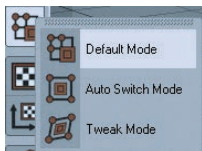
The **Icon Palettes** stretch down the left side and across the top of the editor window. The horizontal palette contains the tools you will be using most often, depending on which module you are using at the moment. If you are using the Modeling Layouts, for example, tools needed to work with polygons, edges and points will be displayed. You can use one of the standard layouts or create your own Icon Palette. CINEMA 4D lets you choose which layout you want to work with.



Now we will concentrate more on CINEMA 4D's icon palettes to get you a little more familiar with them. The following explanation will refer to the colored icons on the next screenshot.



The image shows the left icon palette. At the very top you can see the previously mentioned predefined layouts. Below that we have the (green) "Make Object Editable" button. This function lets a primitive be transformed using points, polygon or edges. The editability of primitives is limited until they are transformed! You can determine size and number of segments but you cannot make any polygonal transformations. Next we have the "Use Model Tool" and "Use Object Axis Tool" (red icons). You can move, scale or rotate a selected object or rotate it around its own axis. The next three (blue) icons represent the "Use Point Tool", "Use Edge Tool" and "Use Polygon Tool". Use these tools to either move, scale or rotate an object's points, edges or polygons or edit the object with CINEMA 4D's integrated tools. The next (purple) icon lets you choose between point, edge or polygon selection.



You select your points, edges or polygons in "**Default Mode**" by simply activating the corresponding points, edges or polygons. In the "**Auto Switch Mode**" CINEMA 4D recognizes whether your cursor is over a point, edge or polygon. A click of the left mouse button selects the correct mode automatically. The "**Tweak Mode**" lets you do the same with an active move, scale or rotate tool. Now you know the icon palette's most important functions.

Now we will turn our attention to the most-used icons on the top palette.

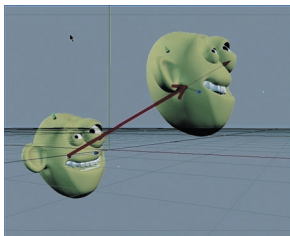


On the left you will find the **“Undo / Redo” arrows (yellow)**. This lets you reverse or repeat each step. You can determine how many steps CINEMA 4D lets you undo by changing the presets in the main menu (edit / preferences / document).

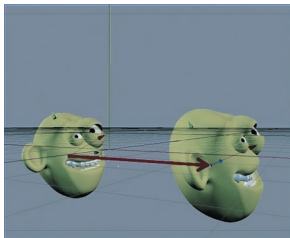
Next to the Undo / Redo icons you will see the **“Live Selection” (pink)** tool. This lets you select your points, edges or polygons for editing. The next three **(turquoise)** icons are pretty much self-explanatory. Use these to move, scale or rotate your object or your object’s selected points, edges or polygons. When rotating, please note that the center of the rotation will always be that of the active object (or camera). The next icon (dark blue) is the **“Selection History”** icon. Clicking and holding on this icon displays the last eight tools used. This makes switching to a recently used tool much easier.

The following three **(red)** icons let you **lock & unlock the X, Y or Z axis**. These settings let you determine the direction in which your object will be edited. If only one of these icons has been activated it will only be possible to move the object in that particular direction, unless you are using the object axis arrows, which are always independent of the locked or unlocked X, Y or Z directions.

The next function **(purple)** lets you switch between the **“Use World / Object Coordinate System”**. Let’s assume the object axis of your wonderfully modeled head is slanted (whereas the term “wonderfully modeled” is open to interpretation in this case ;o). If you lock the X and Y axis, make the head active and move it, you will notice that your model moves in the X-direction of the object axis.



Now select the world-coordinates instead and see how the object moves on the X axis parallel to the world coordinate system.



This function can be very useful in modeling or animating your scene.

Now to the next group of three, the **(green) Render Icons**. The first function (Render Active View), with a clipboard as a symbol, renders the image in the editor view. The rendering will be made using the settings you have specified, with exception of image size and several post effects. Icon number 2 renders the image in the “picture viewer” using the settings you have specified in icon number 3’s “Render Settings”. You can also render animations in the picture viewer since the function “Render Active View” (as the name states) is only meant for checking the scene in the active view.

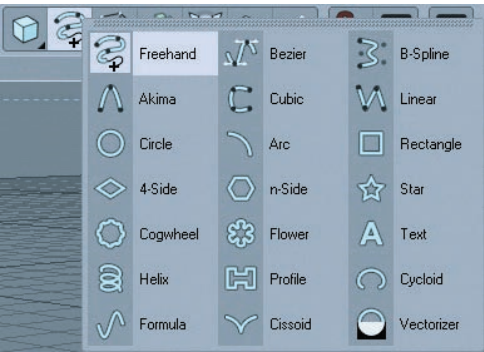
The next **(blue)** symbol is the group window “Add Cube Object”. It contains all of CINEMA 4D’s available predefined parametric objects.



One click and the world’s most used object is created – the cube. Click and hold to see all available parametric objects. This is where you choose the initial shape you will need. And don’t forget! “Only parametric objects that have been converted to polygon objects can be edited at a polygon, point or edges level!”

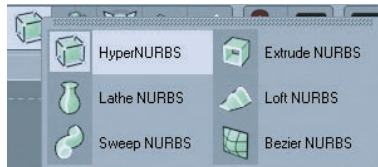


The **yellow** icon represents **spline objects**.



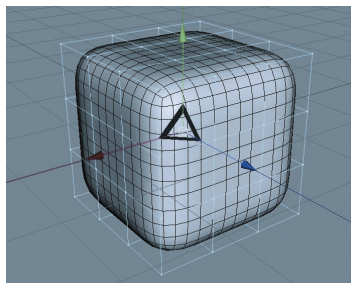
The term “spline” has its origin in ship building. The wooden slats which were elastic enough to conform to the shape of the ship’s hull were called splines. In the 3D world splines can be defined as “point-based curves”. A spline “follows” several previously defined points while still retaining a curved form. This group window offers several tools for drawing splines, as well as predefined shapes from which to choose. A spline can act as a path for a camera to move along. Just draw a spline and let the camera move along its path. Splines can also be used to model. To put it simply, splines are placed in a row as a wire frame over which a skin is stretched using “Loft NURBS”, for example.

The next (**dark orange**) group icon hosts probably the most important CINEMA 4D object, the “**HyperNURBS object**”.

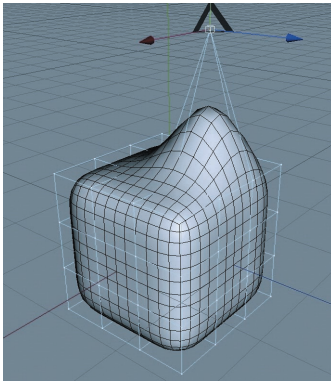


If a polygon object is a sub-object of a HyperNURBS object it will be virtually subdivided to a higher degree. Visually it will be comprised of many more smaller polygons than before the subdivision (the object automatically looks softer / rounder). As you can see in the next screen shot: The outer mesh (light blue) shows the polygon cube’s actual subdivision.

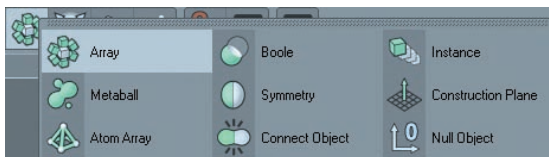
The finer inner mesh (black) shows the subdivision of the HyperNURBS object. Change the cubes’ display mode by selecting (deactivating) Tools / Isoline Editing in the main C4D menu and switching to Gouraud Shading (Lines) in the Editor’s Display menu. In the end it’s up to you how you want your objects displayed in the Editor. However, for this tutorial, this is the most effective way to show the effect HyperNURBS objects have on polygonal objects or primitives since it shows how the cubes are subdivided and the final result is therefore also easier to visualize.



The advantages, especially in modeling, are obvious. Since the object contains few points (edges / polygons) that can be edited it remains very manageable. You can drag just one point of the original wire frame and the HyperNURBS mesh, with its finer subdivision, will follow the point being dragged (see next screenshot).



If the polygon object were made up of such fine subdivision modeling, it would be much more complicated. You would pull one point and only one point would be moved. All other surrounding points would retain their position. You would have to move each one individually in order to achieve the desired shape. Haven't quite understood? No problem, in part 2 of our Quick Tutorial you will try modeling like this yourself so you can learn the functions first-hand. Of course this group window contains several NURBS objects, of which you have already gotten to know the loft and HyperNURBS. Let's take a look at the objects behind the **(green)** "Function Objects".



Here you will find, for example, the **null object** (object without geometry), the boolean object for boolean operations (parametric and polygon objects can cut / slice each other), as well as the symmetry object, which can be unbelievably helpful in character modeling. You simply model one half of the figure and use the Add Symmetry Object function to mirror it and create the other half of the figure. The second-to-last icon contains at least one object without which the best model would appear inconspicuous and flat: the light.



The proper lighting of a scene is at least as important as the scene itself. You can make a better impression with simple models and great lighting than you can with a fantastic model lit by a default light. We'll go more into detail about that later. Here you will find camera objects, the sun object and the environment object, among others (adds a general color and / or fog to the scene). The **"Deformers"** can be applied with the objects of the last icon (light blue).



Use these to bend, deform or squash objects for modeling or animation. There are several helpful tools in this group window. After you have worked your way through this tutorial you can try some of these yourself. You can deduce what most of them do by their names (which appear at the lower left of your monitor when you place the cursor over each icon).

3. Sample Images

This is the “ooohs and ahhs!” section. Take a look at these images and let them inspire you a little before we move on to the hands-on part of this tutorial.



© Milan Soukup



© Joe Yan dr_heyjoe@hotmail.com



© G. Ferrero Moya gferrerom@gmail.com



© Chris McLaughlin



© Anders Kjellberg, www.dogday-design.com



© Kevin Capizzi

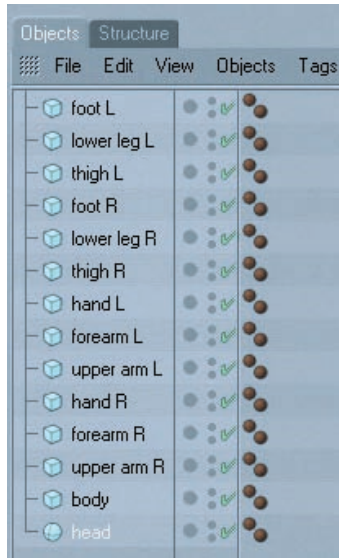
4. Quick Tutorial – Arranging Objects

In order to give you a feeling of how CINEMA 4D works we will begin with the simple creation of a couple of basic objects.

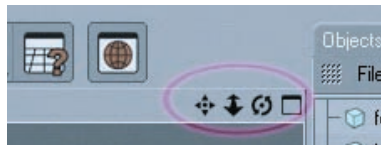
Create 13 cubes and one sphere using the main menu (objects / primitive / cube / sphere) or the group icon “Add Cube Object”. “13 cubes” may give you the impression that we are preparing to create a mammoth project but don’t worry, we are going to arrange the cubes into a little figure. When you’ve created the cubes you can see their alignment in the Object Manager at the right.



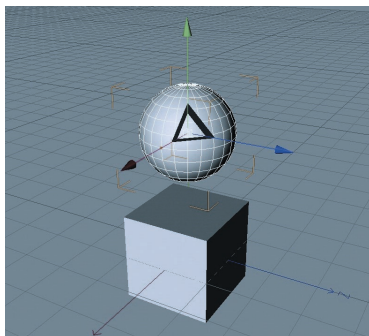
For better reference, give each cube a unique name (double click the current object name in the Object Manager to open the data entry field for renaming the object). You can simply refer to the next screenshot.



As you can see in the editor window, only one cube is visible. That's because all cubes are located at the same coordinates and are the same size, with the sphere in the center. Of course we will want to change this state now, but first a quick introduction to navigating the editor window. How do I rotate and move my point of view? Simple. Take a glance at the top right corner of the editor window. Here you will find four small symbols with which you can change your point of view (of course we mean the point of view of the editor window, not your personal point of view. We can have little influence on the latter.)



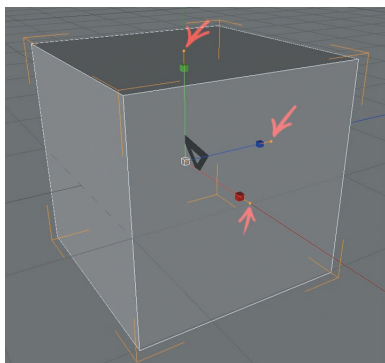
The first symbol (click – hold – move mouse) moves the view. The second symbol (foreshortened double arrow) lets you zoom in and out and the third (curved arrows with a dot in the center) lets you rotate the scene. Selecting the little rectangle to the right will divide the entire editor into four views, giving you a better overall view of the scene. Each of the four views has its own little rectangle which, when clicked, enlarges the respective window. Zoom out a little and select the object “Head” in the Object Manager. The head’s axis will be visible in the editor window. Drag the head’s green object axis to a point over the cubes.



Each of the axis' arrows can be selected and dragged in its respective direction. This prevents the object from being dragged in the wrong direction in the editor view (as opposed to clicking on the object itself and dragging it). It is often impossible to see in which direction an object is being dragged in a 3D view. A similar method of moving an object in a single direction is the previously mentioned locking of a specific axis in the command palette. This prevents an object from being moved in the direction of an axis that has been locked unless you click and drag one of the object's own axis arrows. These objects are not locked. Select the "Body" object and then click on the "Scale" function at top.

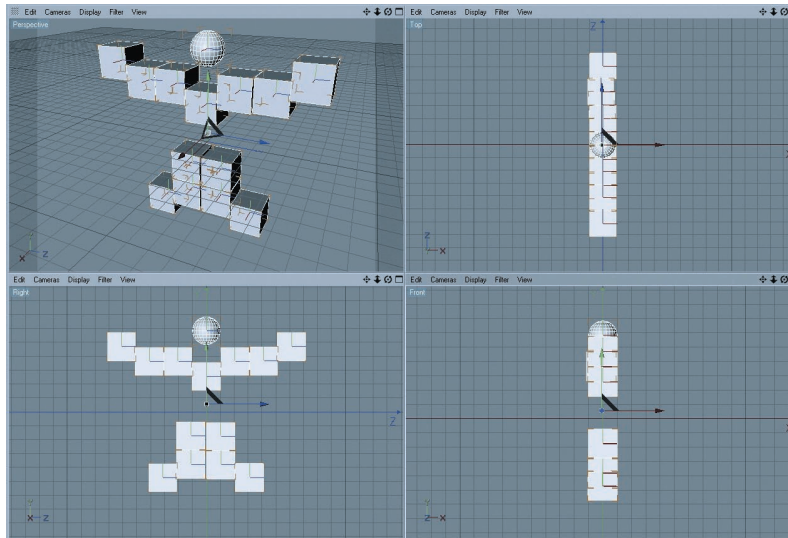


The ends of the axis' arrows have changed from arrows to boxes. Dragging these boxes will scale the object along that particular axis. Parametric objects (not converted polygon objects) will display little orange handles.

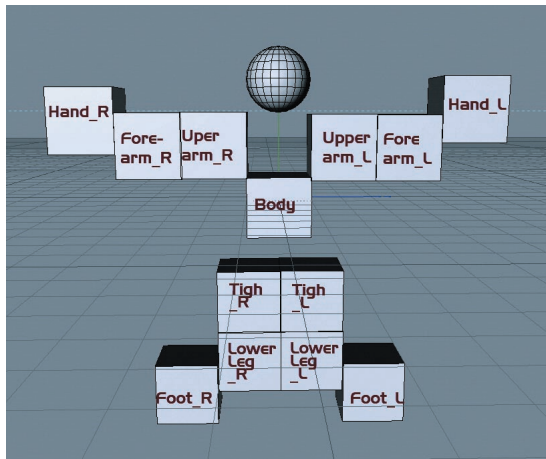


They make it possible to stretch and squash the parametric object on the respective axis

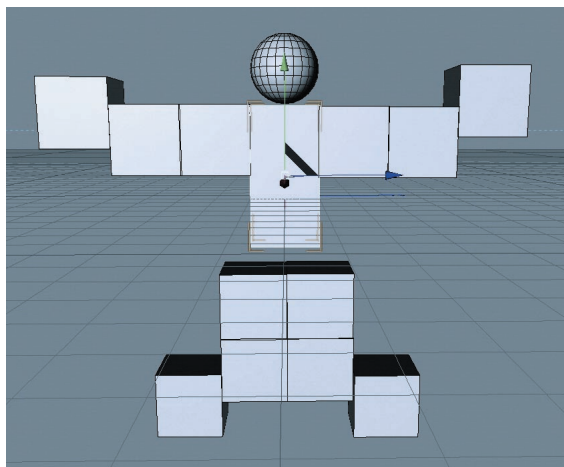
Now we'll get to the nitty-gritty part of this project. You now have enough basic knowledge to be able to arrange the objects according to the following screenshots.



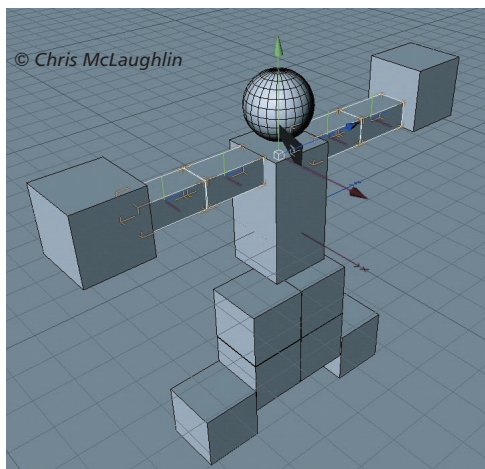
If you prefer, you can switch to a four-editor mode (click the little rectangle at the top right of the editor window). If the objects are displayed as wire frame objects you can switch to “Quick Shading” or “Gouraud Shading” under “Display” in the editor’s menu. Now let’s get to work. Here’s a screenshot of the figure from the front for reference.



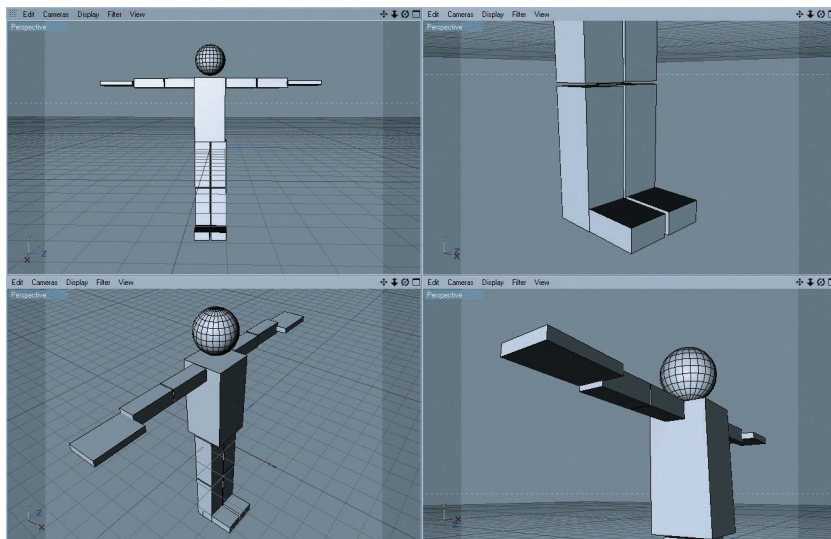
After you have arranged the cubes it still looks nothing like a “human character”. We have to rotate and stretch the figure a little. Click on “Body” in the Object Manager and select the orange handle on the (green) Y axis. Drag this handle until the top edge of the cube lies even with the arms.



Using the orange handles, select the cubes that make up the arms and adjust their size and position as shown in the next screenshots.

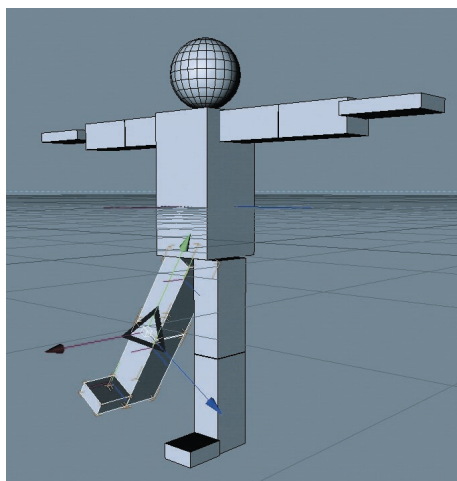


If you want to scale all cubes of the arms at once you can do this as follows: with the shift button pressed, select the objects "Lowerarm_L", Upperarm_R", Lowerarm_R" and "Upperarm_R" and press "C" on your keyboard to convert the objects and scale them with the "Scale" function along the Y and X axis. As you can see, the orange handles are not visible. Squash and move the figures arms and legs until it looks like the following image.



This should be no problem with the knowledge you have gained up to this point. To squash the legs, for example, you can squash several objects at once instead of each one individually (as was the case with the arms).

When you're finished select the objects "Upperleg_R", "Lowerleg_R" and "Foot_R". Once they are highlighted in white in the Object Manager press the "Alt+G" key on your keyboard. This groups all and makes all selected objects children of a Null object. If you look at the Object Manager you will see the newly created Null object. Clicking on the "+" symbol will open the hierarchy and the objects we just selected will be shown. When the Null object is selected, the axis of this Null object serves as the axis for all three leg objects. If this axis is rotated all children of this Null object will be rotated. Move and rotate the Null object a little and you can make the figure stick its leg out.



After you have selected the “Rotate” function you can select the axis rings of the rotation sphere and rotate the object into position. Try changing the figure’s position by using these different “moving” functions. If necessary, group objects into a Null object or select several objects at once in order to bring them into position.



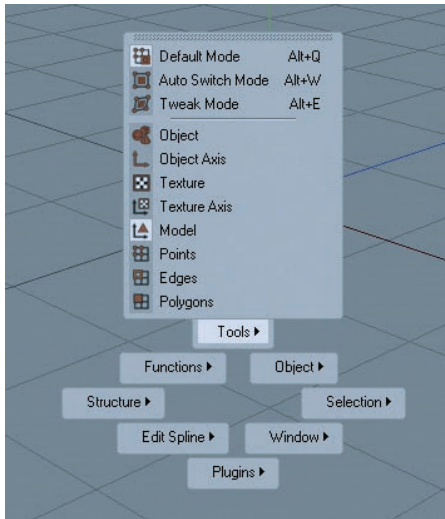


5. Quick Tutorial – Modeling

This is the most important part of this tutorial: How is a model built?

CINEMA 4D R11 has numerous tools that make modeling even easier and greatly simplify workflow.

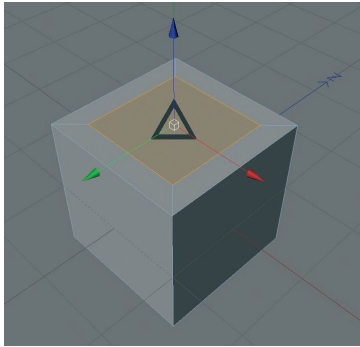
A helpful function for quick navigation is the “General Popup” which you can activate by pressing “V” on your keyboard.



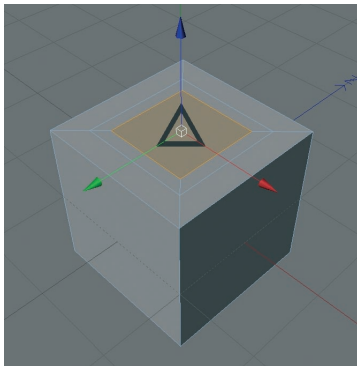
A circular menu lets you choose from several menus in which sub-menus appear when the cursor is placed over them. Play with the menu a little and find out how it can improve your workflow.

In order to show you the basic functions and the most common way to work with the modeler we will create an eye for a comic character.

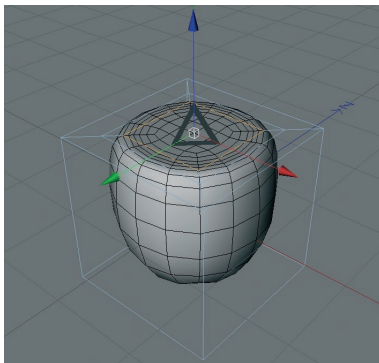
Let's start with the creation of a cube, which happens to be the most-used primitive for modeling (Objects / Primitive / Cube). Press the “C” key on your keyboard. By doing this you have just converted the parametric object to a polygonal object. Most commands can be executed via so-called “hot keys” which, when used heavily, can speed up your work in CINEMA 4D quite a bit. Switch to the “Use Polygon Tool” mode (on the command palette on the left) and select the “Live Selection” tool (upper command palette). Make sure that “Only Select Visible Elements” is active in the Attribute Manager. Mark the cube's top polygon which will become brighter when you place the cursor over it and turn orange when you have selected it. Click on this polygon with the right mouse button. Choose “Extrude Inner” from the menu that appears (hot key “I”). With the left mouse button click on the top orange polygon, hold the left mouse button pressed and drag the mouse a little to the left. A second square should have been created, as seen on the next screenshot.



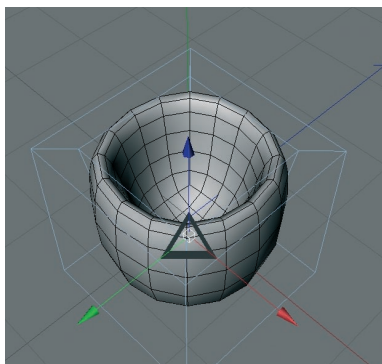
Repeat this procedure to create another square on the top of the cube.



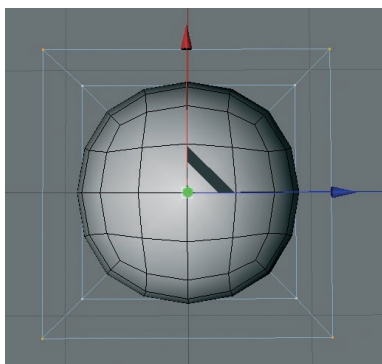
Create a HyperNURBS object using the top command palette and make the cube a sub-object of the HyperNURBS object. This will serve to subdivide our polygon object without us having to subdivide the original mesh. (Select the cube in the Object Manager and drag it onto the HyperNURBS object and let go when the little arrow points down.) Select the cube in the Object Manager and alt+click on the HyperNURBS symbol. This will make your cube a Child of the HyperNURBS object Your cube will now look like this:



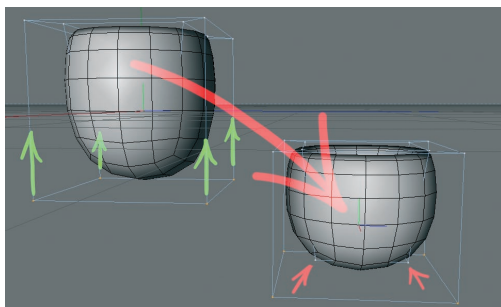
Grab the blue Z-axis in the editor window and drag it down until a relative large indentation has been made.



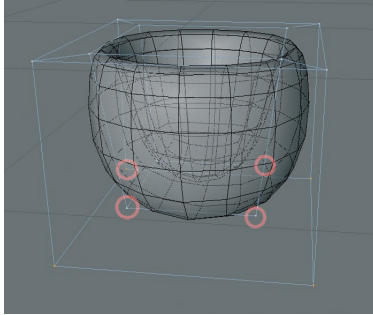
Rotate your view until you have a good view of the underside of the cube and switch to the “Use Point Tool” mode. Now, using the “Live Selection” tool, select all four points on the underside,



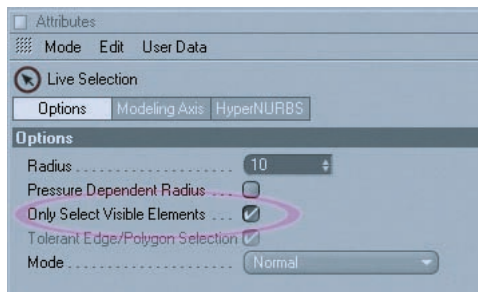
switch to the side view and drag these four points using the green Y-axis – drag them until the four inner points of the indentation can be seen.



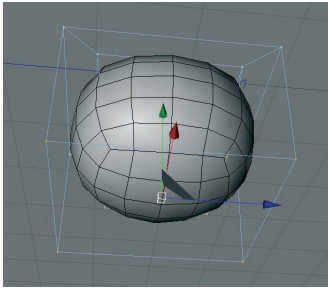
If you created the indentation deep enough you may have already been able to see these four inner points. (In the next screenshot you can see an X-ray view of the cube in which you can see the hidden points very well. More on “X-ray” at the end of this chapter).



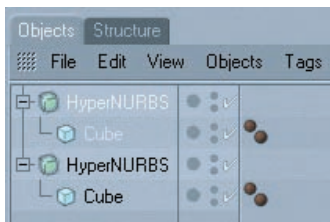
We want to round off the shape a little more and to do that we will select the inner points. Even though they are visible you won't be able to select them with the “Live Selection” tool. This is due to the fact that “Only Select Visible Elements” in the “Options” menu of the “Live Selection” tool in the Attribute Manager is active. Deactivate this option and try the selection again. Now it's possible to select the points. Be careful! If you forget to turn this option off you might select all the points in the front of the object and accidentally select all points at the backside of the object as well. The surface on the backside will be altered and you won't notice until you rotate the object at a later point. So always be aware of this option in the Attribute Manager!



Once you have selected all eight points on the object's underside drag them along the green Y-axis a little to the top to give the object a rounder look.



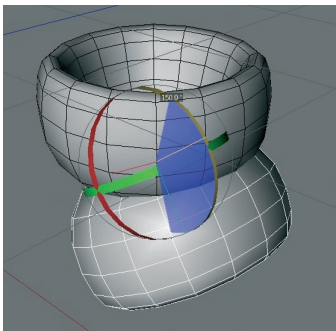
Click on the HyperNURBS object in the Object Manager and drag it down a bit while pressing the “Ctrl” button on your keyboard. We have now duplicated the hemisphere. The same object is now visible in the Object Manager twice.



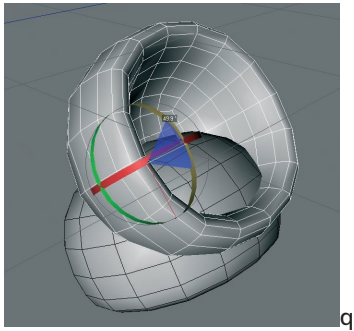
Now select one of the HyperNURBS objects and select the “Rotate” tool.



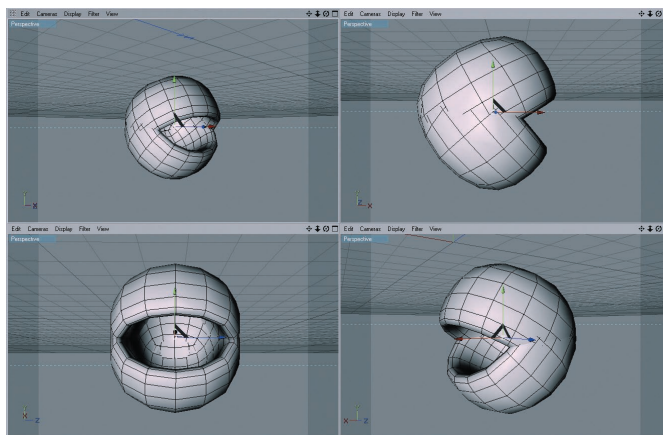
You can now adjust the object’s angle by using the “Rotation Rings” on the “Rotation Ball”. Drag the blue Z-axis ring down 150 degrees.



Repeat this step for the other HyperNURBS object but only to 50 degrees.



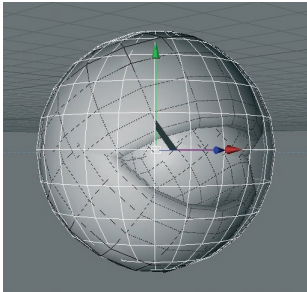
Position both hemispheres as pictured using the “Move” function:



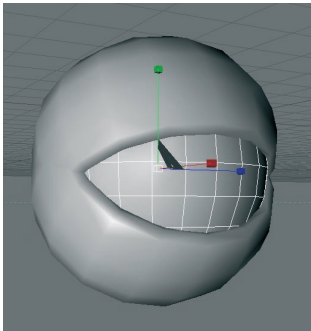
Here you can use the aforementioned locking axis function and switch to using the world / object coordinate system.



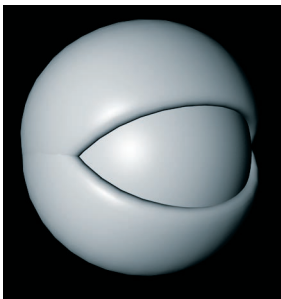
Now create a sphere and move it into a position almost completely covering both hemispheres.



Using the “Scale” function you can resize the sphere to fit inside the two hemispheres.



Congratulations! You have just created your first modeled object.



You can increase the HyperNURBS object’s subdivision to give our model a smoother look. Simply select the respective HyperNURBS object you want to smooth and set the “Subdivision Editor” in the Attribute Manager’s “Object” menu to a higher value. The parameter “Subdivision Renderer” is only responsible for renderings in the picture viewer.

Our eye still looks a little blind. We'll change this in the next chapter by adding a pupil texture. Before we do that, though, we'd like to give you some more modeling tips.

Adjust the influence of HyperNURBS: Select both cubes and several polygons in the Object Manager. Press the "V" key on your keyboard, select the "Structure" menu and click on "Weight HyperNURBS". If you now click anywhere in the editor window with the left mouse button and drag the mouse to the right you can determine the strength of the HyperNURBS for the selected polygons. If you are not satisfied with the result and have unclean edges try this function in the "Use Edge Tool" mode. This will give you better results. You can also simultaneously press the "." key while dragging in the Viewport.

If you should want to select points that lie within an object or if you have "Only Select Visible Elements" deactivated and want to avoid accidentally selecting points on the backside of the object simply activate "X-ray". You will find this function under display / x-ray. This lets you see through the object and see every point (Polygon / Edge). Accidental selection of hidden points is thus not possible and you have an excellent overview of the inner points of the object which would normally not be visible from the outside.

6. Quick Tutorial – Materials

A well-modeled object can make a mediocre impression if the right textures aren't used. Textures give a model color, highlights, structure and other important surface properties. A texture placed into the bump channel, for example, gives the object's surface an uneven, bumpy look without actually altering the geometric structure. This effect can be used to imitate skin wrinkles, scars or the surface of an orange. The displacement channel works in a similar fashion, only that it actually does change an object's geometric structure. Using the luminance channel you can give an object's surface a self-illuminating property or integrate an SSS effect (sub-surface scattering) which lends the surface a slight translucent / reflective look, like human skin or candle wax, for example. In short: Textures have the same significance as the outer shape of an object because they are necessary for achieving the desired atmosphere, coloring and surface structure.

We will begin with a brief introduction to the individual material channels:

Color: This is where the material's color or the base color for the texture is set.

Diffusion: This channel makes your texture "irregular". Through the application of a noise shader or a texture your object receives a dirty or dusty look. If desired it can also influence the specular, reflection and luminance channels respectively.

Luminance: The material is given an illuminative property which is also taken into account in the Global Illumination calculation.

Transparency: This is where you determine the material's opacity.

Reflection: Gives the material reflective characteristics.

Environment: A texture is used to simulate an environment reflection.

Fog: This channel lets you apply a fog property to a material.

Bump: Uses an optical trick to translate light and dark elements of a texture or a shader to simulate the height and depth of an uneven surface. Scars, wrinkles or scratches can be simulated using this channel.

Normal: This channel is meant for use with “normal textures”. Normals give a low-res polygon object a hi-res look when RGB textures containing the required properties are applied. This lets a hi-res polygon object be replaced by a low-res object, thus saving a lot of render time and offering the same visual result.

Alpha: A texture’s transparency is determined by a material’s light and dark areas. Black equals a transparency of 100% and white makes it opaque.

Specular: This determines a material’s specular properties.

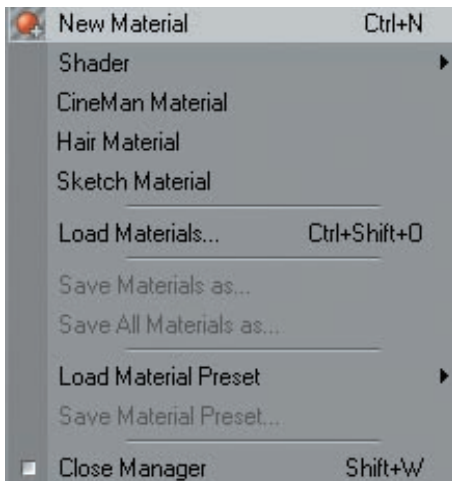
Specular Color: This determines the color of the material’s specularity and can be influenced by a texture.

Glow: Gives the object a self-emitting glow.

Displacement: Deforms an object using light and dark values (calculates differences in height). Do not confuse this with the Bump channel which only imitates an uneven surface.

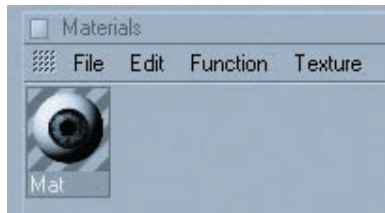
Since our eye still looks a little pale we will liven it up a little with the application of textures and shaders. Open the “QS_Material.c4d” file. Now we have the eye we created in the previous chapter. You can see in the Object Manager to the right that the object does not yet have a texture applied to it. We will do something about that now.

Click on file / new material in the Material Manager at the lower left.



A standard material has been created. If you click on this material its properties will be made visible in the Attribute Manager to the right. In the “Basic” menu you can determine which channels should be activated for this material. Go ahead and activate the Bump channel. As soon as you have done that a new menu button, “Bump” will appear. Now click on the menu button “Color” and load a texture into the material by clicking on the small arrow next to “Texture”. Choose “Load Image” and load Iristexture.jpg.

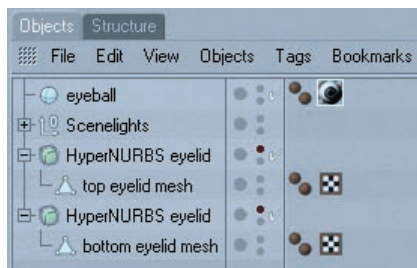
In the mini-preview of the Material Manager at the lower left of your screen you will see the texture displayed as soon as it has been loaded. This gives you a good overview of the materials being used in the scene.



Repeat this procedure for the “Bump” channel and load `Iristexture_bump.jpg` into the channel. This JPEG contains the gray scale version of the iris texture which we need to create a relief effect for the surface. You can also choose “Filter” (click on the small light gray arrow next to the word “Texture” in the Bump channel) and load the color texture here and set its saturation to 100%. This saves you from having to load a second image. The bright areas of the image will later appear to be raised on the object and the dark areas of the image will appear to be somewhat indented. A true deformation of the object will only take place in the “Displacement” channel. The “Bump” channel does not alter the polygon’s surface but uses an optical illusion to give the surface its structure.

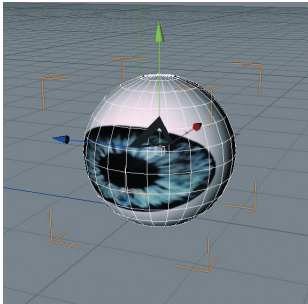
Click on the material in the Material Manager with the left mouse button and drag it onto the object eyeball in the Object Manager. (When you drag the material over the object you can let go once the little black arrow points down). Alternatively you can drag the material onto the desired object (the eyeball) directly in the Editor. Just make sure you drop the material onto the correct object if there are several in the scene or in close proximity to one another. You can check in the Object Manager to make sure the material was dropped onto the correct object - the material icon will appear next to the object onto which it was dragged.

You have probably noticed that the eyeball brightened somewhat after you applied the material but you aren’t able to see the actual texture. We still have to change the offset properties and the mapping size so the texture will be aligned properly on our object. At the moment the actual image of the iris is lying distorted on the left side of the eyeball. You can check this by making both HyperNURBS eyelid objects invisible for the editor. To do this double-click on the top small gray dot to the right of the object in the Object Manager (until it turns red).

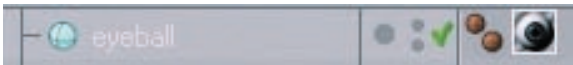


Double-click on the dot again and it will turn green, which makes the objects visible again independent of the visibility settings of any parent object. The dot directly below has the same function except that it affects the rendering.

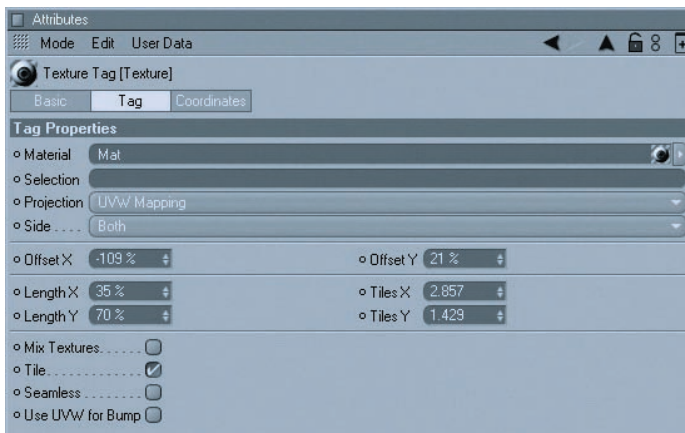
Once you have made the eyelids invisible and have rotated the view a little the eyeball should look as follows:



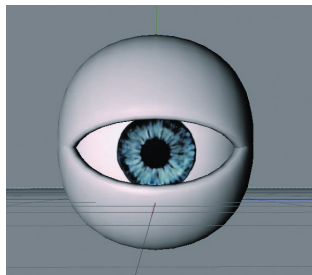
Switch the visibility of the HyperNURBS objects back by clicking again on the dots next to the object in the Object Manager, making them gray. Click on the “Texture Tag” at the right of the Object Manager next to the object. It’s the material that we applied to the eyeball. You can recognize it in the mini preview of the texture in the Object Manager.



Once you have selected it you will see its parameters in the Attribute Manager. Adopt the settings you see in the next screenshot:



We have just aligned the texture on the eyeball mesh by changing the “Length X” and “Length Y” parameters. The offset setting put the texture in the correct position. If you rotate your view again you will see that the iris texture is positioned correctly.

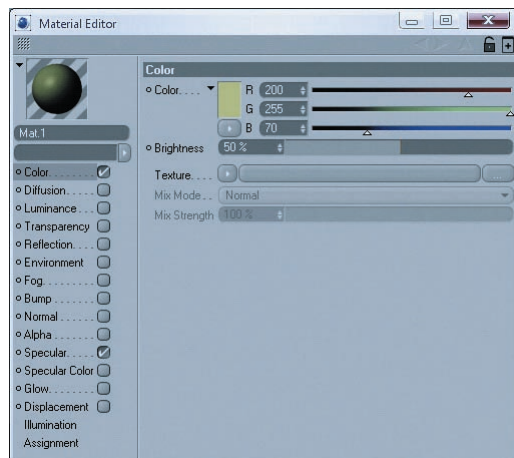


Tip: If you want to undo an accidental change to the view just press “Ctrl+Shift+Z”. This function is useful if you have inadvertently rotated the perspective view instead of the editor view. You can also select edit / undo view in the main menu of the editor view.

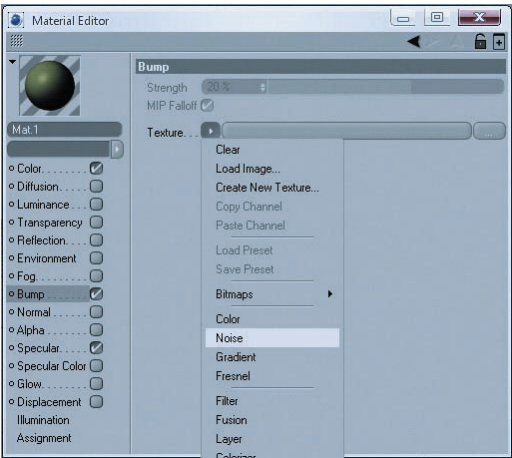
Our eye may be able to look at us now but the eyelids still make it look a little too gray. We will change a couple of settings that will give the eye a reptilian look.

Create a new material (Material Manager / File / New material) and double click the new material. This will open a dialog window for the material where we can make the necessary changes to this material. Click on “Color” in the material channel and copy the settings of in the following screenshot.

Click on “Color” in the material channel and copy the settings in the following screenshot.

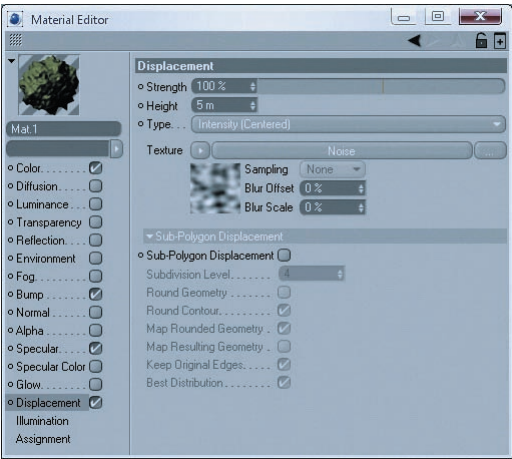


We will give the material a green tone and lower its brightness to 50%. Check the box next to the “Bump” channel. Click on the little light gray arrow in the check box next to the word “Texture” and select “Noise”.



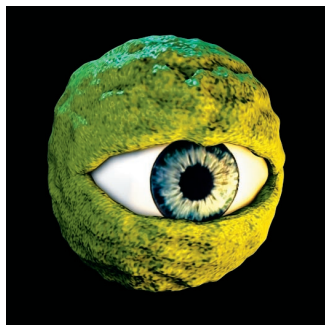
Click on “Noise” and on the following dialog page set the global and relative scale factors each to 30%. This reduces the size of the bump-noise mapping which will result in a finer depiction of the bump map.

Check the box next to “Displacement” and repeat the previously mentioned steps for the bump channel but set the global and relative scales each to 150%. This will increase the size of this channel. Click on the word “Displacement” to return to the displacement channel’s main menu.



The displacement channel deforms the polygon mesh according to the bright and dark areas of an image. Bright areas of the texture raise the polygon mesh and dark areas lower the mesh. This lets you create a wide variety of shapes without having to model such a complex surface, thus saving you a lot of time. The ornamental facade of a house or the relief of a sword handle are good examples. The possibilities are endless.

Close the Material Editor window and set the HyperNURBS subdivision of the eyelids to at least 4 in the editor (Click on the respective HyperNURBS object and change the settings in the Attribute Manager) and apply the new material to the eyelid objects. Render the view (Ctrl+R). The result should be at least somewhat similar a reptile's eye.



You have seen how you can get quick results without having to create a complex texture. CINEMA 4D's integrated shaders and channels offer so many possible variations that you will never be able to try them all. Play around with some of the parameters, add a couple of channels and find out how they influence your renderings.

Here are some more tips about channels for you to try:

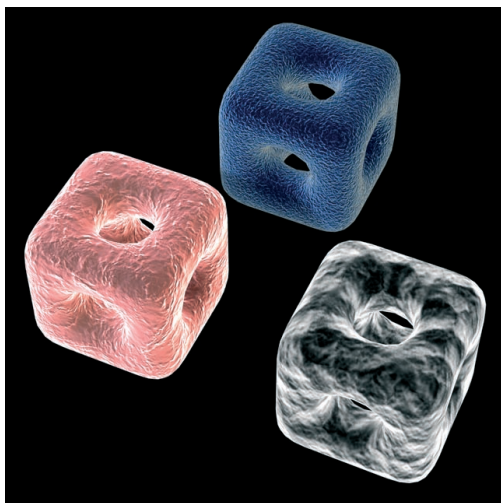
Got dirt? CINEMA 4D does! Most objects in the real world are not as clean and immaculate as they might appear in CINEMA 4D. Real stone figures show signs of weathering over the years and dirt has settled in the wrinkles and cracks. You can simulate such "dirt" very easily with CINEMA 4D (if you own the Advanced Render module) by selecting the "Ambient Occlusion" in the main page's "Effects" menu..

If you own the "Advanced Render" module (or are testing the CINEMA 4D demo version) you can render human skin, for example, very realistically. The Sub Surface Scattering makes it possible. By placing this shader in the luminance channel (effects / sub-surface scattering) the effect is created when rays of light meet a slightly transparent object. Some rays infiltrate the object further and are dispersed, others are directly absorbed or bounce off. Further possible uses for this effect would be for materials such as plastic, milk, candle wax or figurines made of jade.

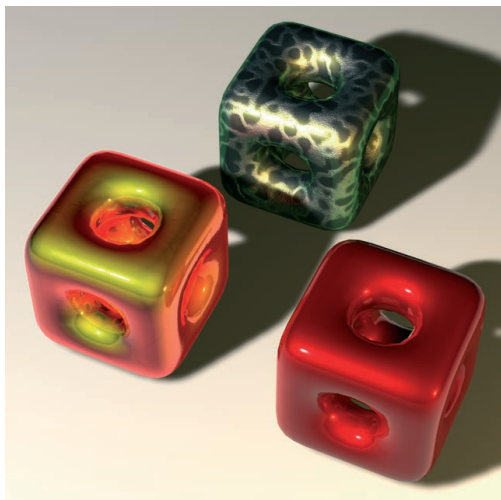
You can load black & white textures into the **alpha channel** to influence the material based on the texture's brightness, similar to the way you would use them for the bump or displacement channels. The texture's black areas would be rendered with a transparency of 100%. As the texture becomes brighter the transparency is reduced accordingly. White would have a transparency of 0%

If you choose "Shader" instead of "New Material" under "File" in the Material Manager you will see a list of 3D shader presets. The advantage of these shaders is that you don't have to worry about mapping your texture or seams in your texture because a 3D shader will be calculated for the 3D space. Here are a couple described in detail:

Cheen: Generates an electron microscope effect good for the depiction of bacteria or mites.



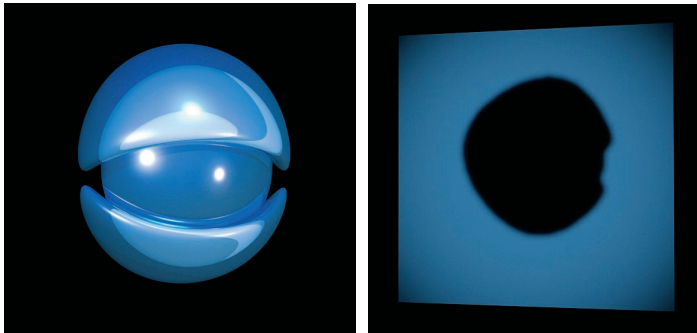
Danel: Very good for simulating high-gloss finish



Banzi: Lets you depict various types of wood.

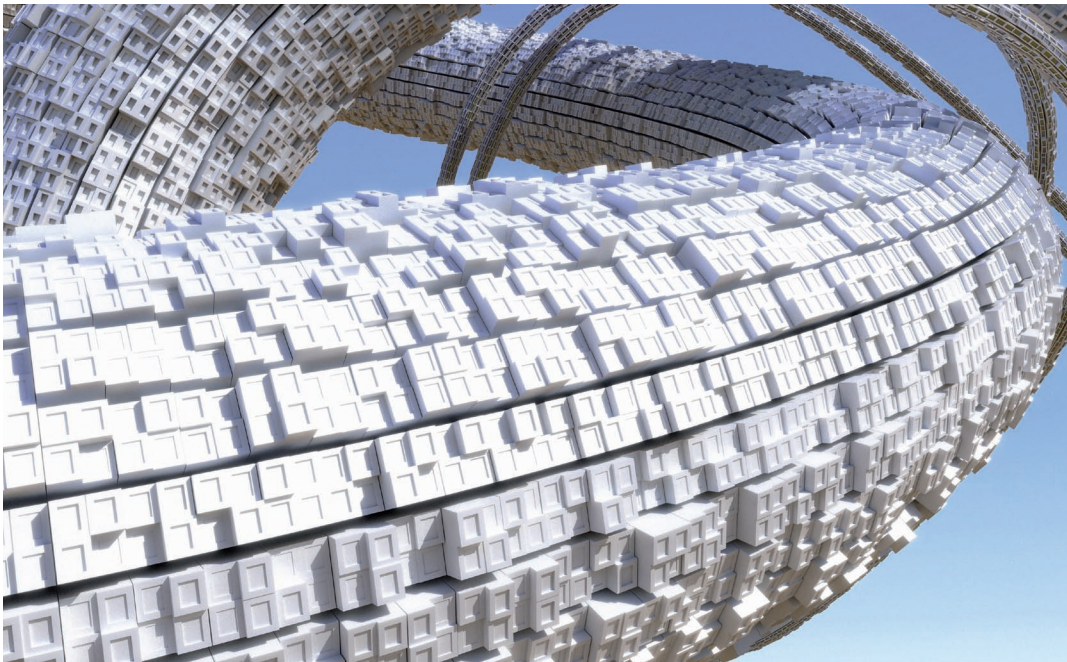


Banji: Calculates complex lighting situations with glass and even makes rear-projection (shadow casting) on partially transparent materials such as rice- or canvas paper possible.





© christoph mensak, brown_eye_architects@gmx.de



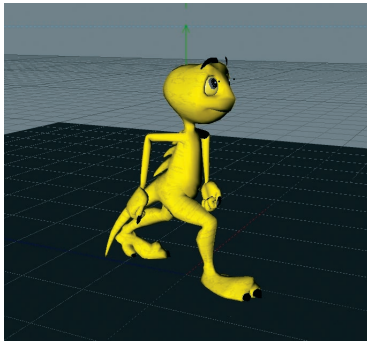
© www.c4d-jack.de

7. Quick Tutorial – Lighting

If you are already familiar with lighting a scene in the “real world” then you will feel right at home with the CINEMA 4D light objects. They can do everything “real” lights can do – and quite a bit more. In this tutorial we will set up a 3-point lighting arrangement. This type of arrangement is used often in portrait photography to achieve an even lighting and is an excellent method for lighting an object quickly and professionally in the 3D world

Adjust your editor view so the entire figure is visible to you.

We want to light up our little character. Open the file “QS_Light.c4d”. Create a floor object (Objects / Scene Objects / Floor) and position it so the figure is standing on it.

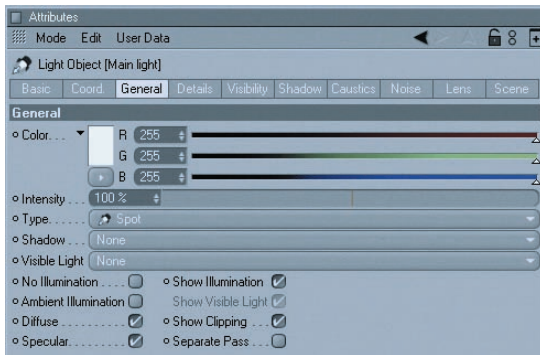


A 3-point lighting arrangement begins with setting a key light. As the name suggests, this light emits the main lighting for the scene and will cast the main shadows. Create a light object (Objects / Scene / Light). Name it “Main_Light” in the Object Manager.



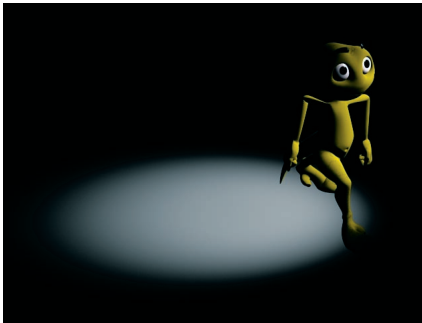
CINEMA 4D has several different types of light sources. The key light will always be created by default. A point light emits from its center in all directions. For our key light we will need a spot light which we can aim directly at the object.

To make the key light a spot simply go to the Attribute Manager and switch the light from “Point” to “Spot”.



Now our light source has been transformed to a spot. A spot acts like a flashlight. CINEMA 4D offers spots with square and round cones of light. This cone is visible in the editor and can be manipulated. Now we will aim the spot at our figure.

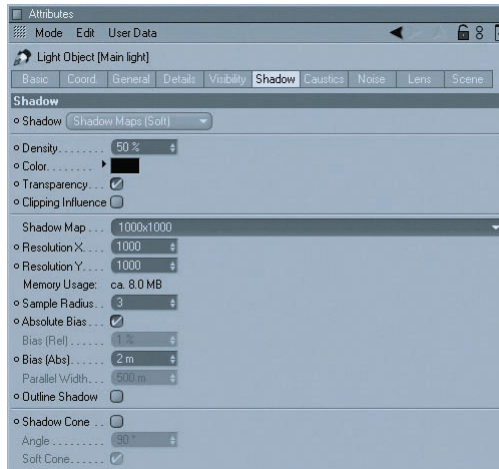
Position the light at the following coordinates in the Attribute Manager: X=300, Y=580, Z=-300 at an angle of H=45, P=-45 degrees. Render the scene.



The light now falls at an angle onto our object (If this is not visible in the Editor it may be due to the fact that your display mode is set to “Quick Shading” (uses a single default light source) instead of “Gouraud Shading” (uses all scene lights)). Of course the exact position of the light is strongly dependent upon the camera’s angle. Unfortunately the light is not casting a shadow, letting the figure look like it’s floating. CINEMA 4D’s lights have an advantage over real light in that you can choose which kind of shadow, if any, they should cast - a plus for any studio photographer.

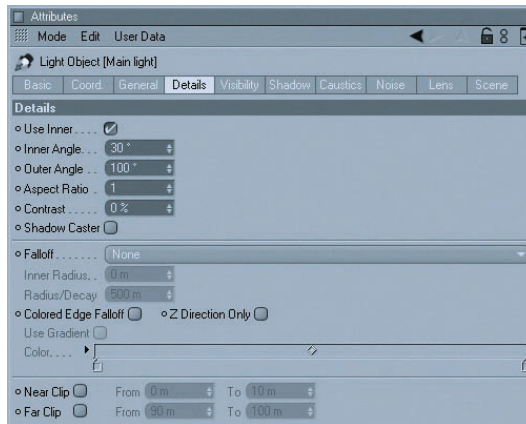
In the “General” menu of the Attribute Manager, set the light’s shadow to “Shadow Maps (Soft)”. We don’t want the shadow to be completely black so we’ll make it a little transparent.

In the “Shadow” menu, set the shadow density to 50%. Select “1000 x 1000” as the shadow map. Render the scene.

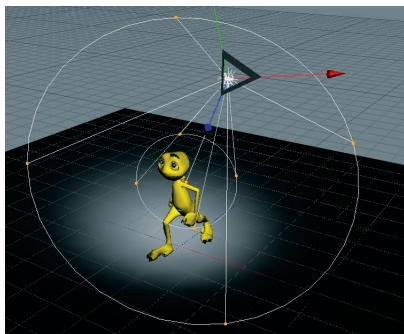


CINEMA 4D offers three types of shadows: “RayTraced (Hard)” – a shadow with sharp edges, “Shadow Maps (Soft)” – a shadow with soft edges and “Area” – a shadow that becomes softer the further it’s away from the object, resulting in the most realistic shadow effect. Try the other two shadow types. *Careful, the area shadow can take a long time to render!* The larger shadow map allows the shadow to be rendered more accurately.

The light’s cone is a little too small. We will change this as follows: Switch to the details menu in the Attribute Manager and set the “Inner Angle” to 30 degrees and the “Outer Angle” to 100 degrees.



You will see the result in the editor right away. You can also edit the light's cone by dragging the orange handles (If your graphics card will support it you can set the editor's display mode to "Enhanced Open GL" with activated shadows. Generally speaking, OpenGL offers a much more precise depiction of your scene and gives you an impression of how the shadows will fall).



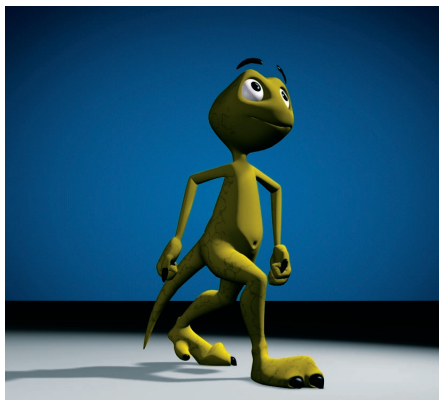
Now we're happy with our key light. Next we will create a more even lighting by brightening our figure a little from the other side.

Create another light source in the scene and name it "Brightener". Place it at the following coordinates: X= -360, Y=225, Z=-230 and at an angle of H=-20, P=-10 degrees. Select "Area" as the type of light.

Since the brightness of the lights in the scene is additive, we must "dim" the brightener a little.

Reduce the "Intensity" in the "General" menu to 40%.

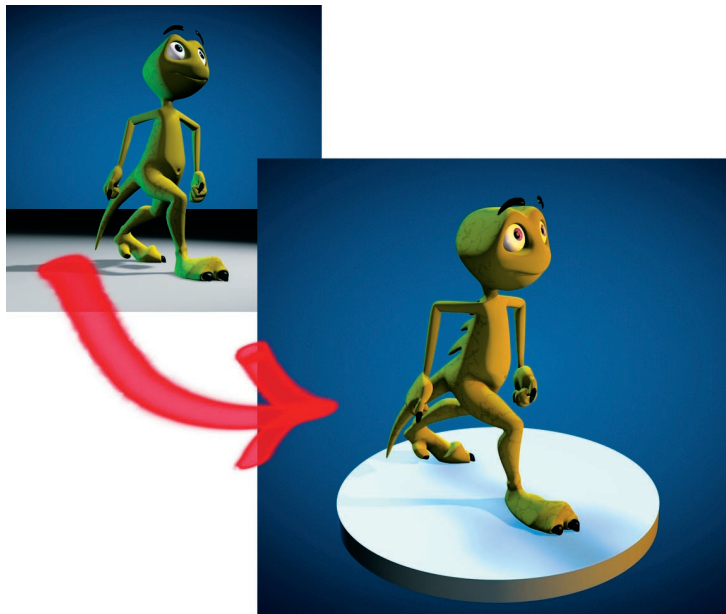
This area light illuminates the figure from a different angle and softens the contrast somewhat. It won't cast a shadow since this would cause "crossing" of the shadows and make the object look bad.



The scene is now pretty evenly lit, but we want to give it a little more pep. Create another light source, name it “Color” and, in the Attribute Manager, set its type to “Infinite”. Set its color to turquoise and set its H angle to -160.

The position of an infinite light is irrelevant since it always lights your scene in the direction of the Z axis. This is why we will leave it at the point at which it was created. It gives our Amphibian an interesting color edge and sets him off of the background a little.

Your scene’s mood can be changed by simply changing the color of some of the lights used.



That completes our classic 3-point lighting arrangement. Now the real work starts. If the scene has a background, which is often the case, it will have to be lit as well. With the proper use of point lights details in the scene can be “brought to light” very nicely. But don’t overdo it. With good lighting, less is often more. Only add lights when necessary and if the scene can actually benefit from them. Two more tips before we end: If you have several lights in a scene and are not sure which light is lighting what, simply make all other lights invisible in the Object Manager. The light which remains will be the only one visible in the editor.

There is a trick how you can determine how to best light which objects in your scene. Select the desired light in the Object Manager and activate Link Active Object in the editor view’s Cameras menu. Selecting this option lets you view the scene from the point of view of an active object, in our case the light. Moving the editor view will automatically change the position of the light when in this mode. This way you can see how the change of position of the light affects the lighting of the object in realtime (Gouraud Shading must be active in the editor view). Once you have reached the desired angle and position you can return to the editor view by selecting Editor Camera from the Cameras menu.



8. Quick Tutorial – Animation

With but a few exceptions CINEMA 4D lets you animate every attribute of an object. This means you can alter any attribute in the Attribute Manager over time, whether it's an object's Y-coordinates, the color of a light or the strength of an explosion object. By animating different attributes you can easily add complex animation effects and visually attractive scenes.

Let's look at a "quick & easy" example just to demonstrate the basic principles of animation.

Begin by opening a new (empty) scene. Create a cube (Objects / Primitive / Cube).

You will see a turquoise slider at the bottom of the editor window next to which the frame (time) is shown. This is known as the time slider. By moving this slider you can jump to a different point (time) in the animation, similar to fast-forwarding or rewinding a film. You can also use the turquoise arrows to the right of the slider to play the film at a predetermined speed.



Further to the right you will see the "record" button (the red button furthest to the left with the key icon). You can use this button to record certain object attributes. Use the buttons to the right of the record button to set these attributes. With these buttons you can "key" (record) the position, size, rotation, attribute and / or point-level-animation of an object at any given time in the animation.

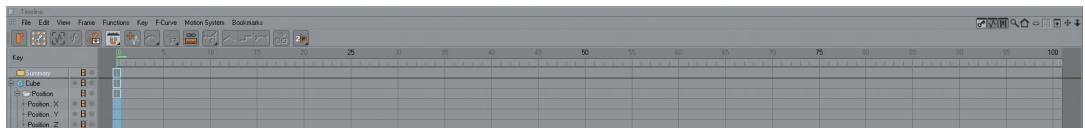


Make sure the time slider is to the left, on 0. Deactivate all symbols to the right of the red buttons, except the first (position), and click on the record button.

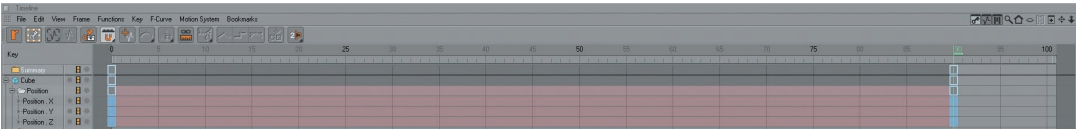
We have now told the cube that it should stay at its position of 0 / 0 / 0 starting at time / frame 0. In other words, we have generated a key that contains the information on the position of the cube at time / frame 0. We will tell you later what exactly a keyframe is. Where can you find this ominous key? It's located in CINEMA 4D's "timeline". The timeline is where you can change the position of the keys on the timeline, change the values they contain, delete them, set new keys and much more.

Switch to CINEMA 4D's animation layout (Window / Layout / Animation).

Take a look at the timeline at the bottom of your screen. You will see the cube along with a "track" for its position. This track contains three "sequences" (one for every recorded coordinate) with a light blue box at time / frame 0 – a key.

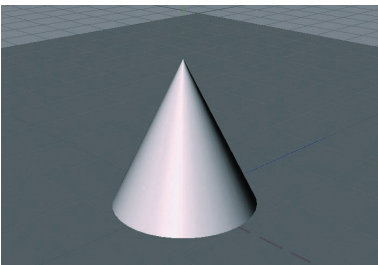


Slide the time slider to frame 90. Move the cube along its blue Z-axis (back) a little. Click on the record button. Three more keys will appear on the timeline, this time at frame / time 90.



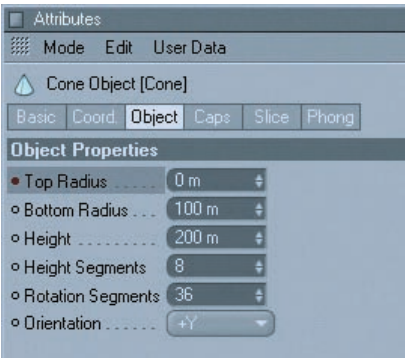
When you move the time slider you can see the cube move between the two recorded points. Congratulations, you’ve animated the cube! Using the red button to record an object’s changing attributes is the quickest and easiest way to generate keys. There is a disadvantage, though. Often, altered attributes will be recorded even if they had not been altered at all. In the case of the cube it was the X and Y positions. There are other ways in which animation keys can be set. We will now look at how you can select and animate specific attributes.

Open a new (empty) scene. Create a floor object (objects / scene / floor) and a cone (objects / primitives / cone). Move the cone up a little along its green Y-axis so that it’s standing on the floor.



The cone has a lot of attributes that we can change using the Attribute Manager. We will now animate two of these attributes - the upper radius and the number of segments of the cone.

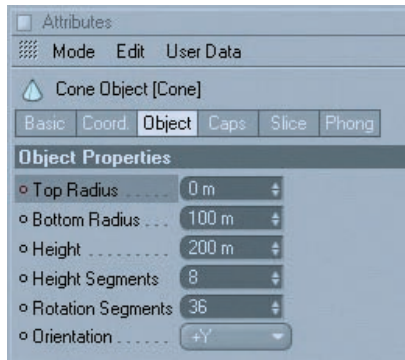
Make sure that your scene is set to frame 0. Hold down the Ctrl key and click on the small black circle in front of “Top Radius”. It will turn red.



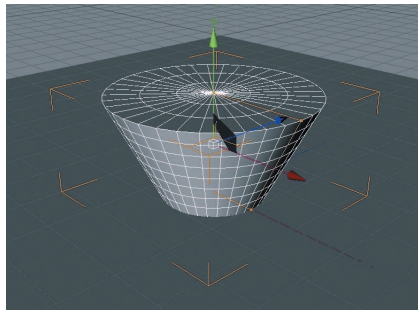
We just told CINEMA 4D that the “Top Radius” attribute of the cone at point 0 of the animation should have a value of 0. Of course we haven’t created an animation yet, only a starting point for the animation. The filled red circle in front of the attribute name tells us that a key has been set at this point in time in the animation. This is an easy way to see if an attribute has been animated. A further CTRL-click on this circle would delete the key.

Go to frame 50.

The filled red circle is now empty. This means that the attribute has been animated at some point in the timeline but no key exists at this particular point.

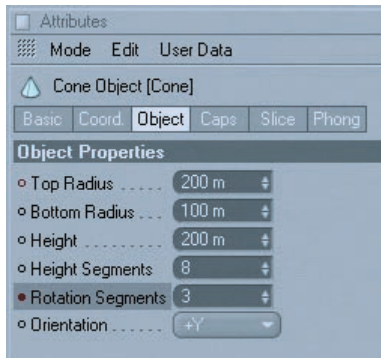


Change the “Top Radius” value to 200 and set a second key using the method described above. Play the scene in the editor and watch how the cone and the value in the Attribute Manager change.



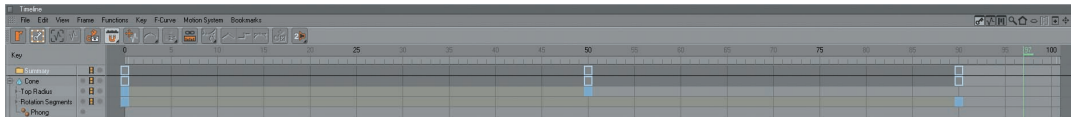
The cone now knows that at frame / time 50 the top radius has to have a value of 200. All changes to this attribute between frames 0 and 50 will be “interpolated”. Interpolated means that CINEMA 4D automatically calculates the necessary values in between. This means the value at frame / time 25 of the animation will be exactly 100 since half the time equals half the altered value of the attribute. The attribute’s value changes over time – it has been animated.

Go back to frame 0. Set a key for the “Bottom Radius” attribute. Go to frame / time 90. Reduce the number of segments to 3 and set another key. Play the animation.



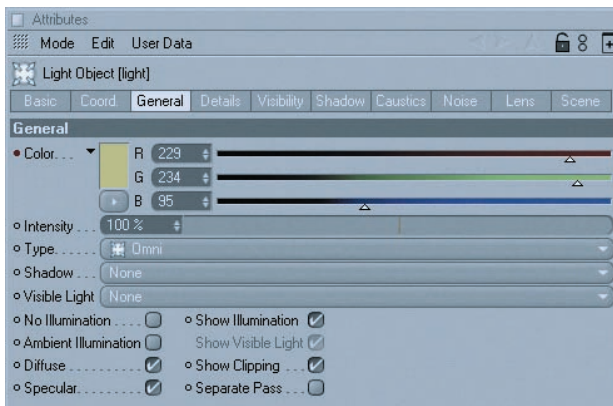
Now two of the cone's attributes have been animated. One changes between frames 0 and 50 and the other between frames 0 and 90. For an overview of the keys that have been set we use the timeline.

If the timeline is not yet visible, simply open it (window / timeline) or switch to the predefined animation layout (window / layout / animation). In the timeline you can see the cone, its animated attributes and the keys that were set.

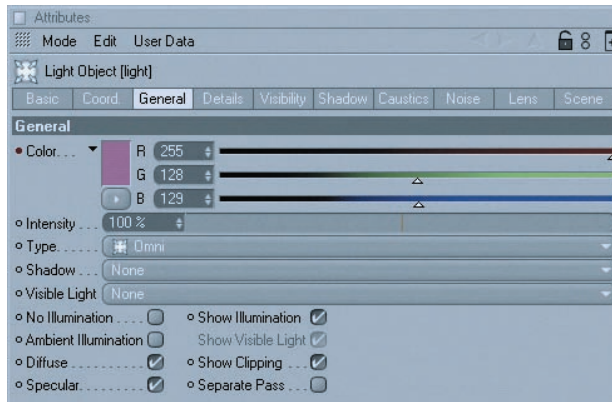


At the end of the animation the cone looks completely different. A nice animation such as ours deserves dramatic lighting so let's not dwell on the cone anymore and move on to other objects.

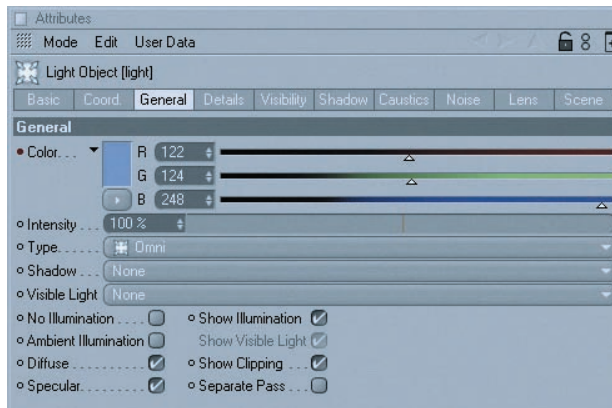
Make sure you're at frame / time zero. Create a light source (objects / scene / light) and position it at X=200, Y=250. Set the color to a light yellow in the "General" menu of the Attribute Manager. Set a key for the color value by CTRL-clicking on the black circle in front of "Color".



Go to frame / time 50 and set the color to pink and set a key.



Now go to frame / time 90, set the color to a friendly blue and set a third key. Play the animation.

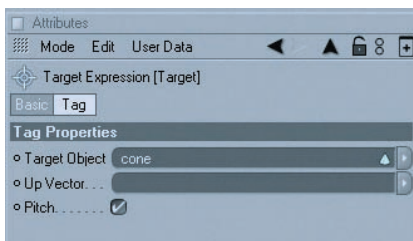


The cone metamorphosis now takes place in animated light. The color sliders change as the animation plays and CINEMA 4D interpolates the colors between the keys that were set. Our animation is becoming more and more interesting. That's why we don't want to see it from only one perspective, but we want to let a camera fly around it.

Create a target camera (objects / scene / target camera). Rename the camera in the Object Manager to "Animated Camera".



A target camera differs from a normal camera in that it focuses on a specific object. This has the advantage that the camera can be moved freely within a scene, without “losing sight” of your targeted object. When a target camera is created, it places with it into the scene an additional object, a Null Object named “Camera Target. 1”. This is the object upon which this camera is targeted by default. In this case we don’t need it because we want to target the camera onto the cone. Delete the “Camera Target. 1” in the Object Manager. Click on the cross hairs next to the camera. Now drag the cone from the Attribute Manager into the “Target Object” field in the “Tag Properties” below.



In order to view the scene from the new camera’s perspective simply switch to the camera “Animated Camera” in the editor menu under “Cameras”. The camera is looking at the scene from above but is not living up to its name since it’s not animated – yet. We would like to animate the camera along a circular path around the scene. For this we will need a circle. Create a circular spline (objects / spline primitive / circle). Set the radius to 1000 and the plane to “XZ”. Set the spline’s coordinates to Y=20 so the circle is above the floor and angle the spline a little by setting the P-angle to 30 degrees.

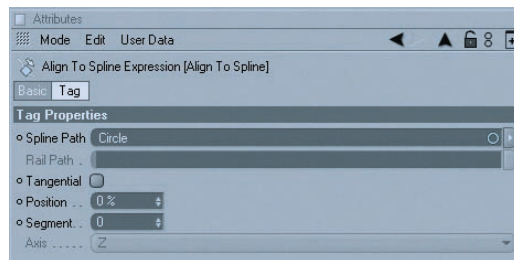


Now we have a circle that encompasses our scene – a track for our camera. We just have to tell the camera to follow this circle. This is done with a special CINEMA 4D tag.

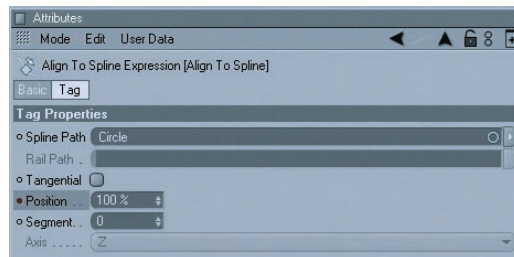
Using the right mouse button, click on “Animated Camera” in the Object Manager and select “CINEMA 4D Tags / Align to Spline” from the menu that appears. A further tag will appear next to the camera.



This tag has a text field named “Spline Path” in the Attribute Manager. Drag the circle from the Object Manager into this text field.

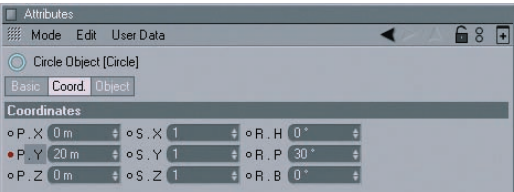


Now the camera is positioned on the circular path. If we play the animation, though, we will be disappointed because the camera doesn’t move even one millimeter. That’s because we haven’t yet told it to move along the circular path. Make sure you set the animation back to 0 and the circle is the active object. In the Attribute Manager under “Align to Spline” you will see an attribute called “Position” which is set to 0% by default. Set a key for this attribute. Go to frame 90 and set this attribute’s value to 100%. Set another key and play the animation.



The attribute “Position” determines how much of the path, in %, will be completed within a given amount of time (frames). This means that the camera followed 0% of the path at frame 0 and 100% of the path at frame 90. It followed the entire length of the path once.

Make sure you set the animation back to 0. In the “Coordinates” menu of the Attribute Manager, set a key for the Y position.



Go to frame 90. In the editor, move the circular path up along its green Y-axis until the camera has a bird’s eye view of the cone. Key the new Y position.

Now the circular path and the camera move together on the Y-axis in the course of the animation. With just a few animated attributes we have managed to create a complex animation.





© Greg Brotherton, www.brotron.com

9. Quick Tutorial – Rendering

You’ve been a busy bee. You have created a scene, set up the lighting, animated objects and assigned materials to them. Now we want to see the result of all this work. What you have to do is to transform this 3-dimensional scene into a 2-dimensional image (in the case of an animation it would be an entire series of images. We will “render” the images. CINEMA 4D offers a wide variety of options for rendering your 3D scene. We will use a scene from the animation chapter as our source and we will add a transparent sphere and a bright background to the scene.

Open the scene “QS_Render.c4d” (file / open). Slide the time slider to frame 20 and click on “Render / Render View”.

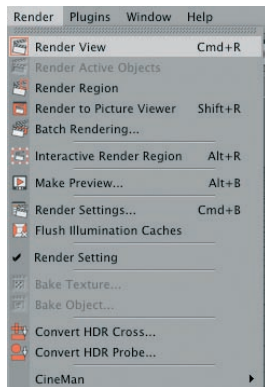


We will render frame 20 using CINEMA 4D's standard settings to give you a quick impression of how our final scene will look. This method of rendering is probably the most widely used since it can be used to make sure the scene "is on the right path".

Slide the time slider to frame 0 and press "Ctrl+R" on your keyboard.

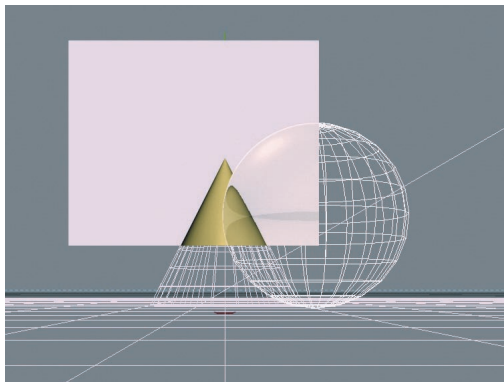
The view has been rendered again. There are three ways you can render the active view.

1. Using the main menu
2. The keyboard shortcut "Ctrl+R"
3. By clicking on the icon in the editor window (the clapboard farthest to the left)



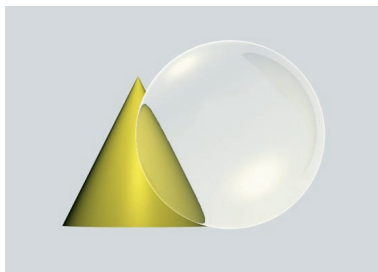
Use the method with which you feel most comfortable.

Often we don't necessarily want to render the complete editor view but only a small part of it. This is also no problem. Select "Render / Render Region". The cursor will be transformed into a cross. Drag a frame around the region you wish to render.



The second possibility is to render only a single object.

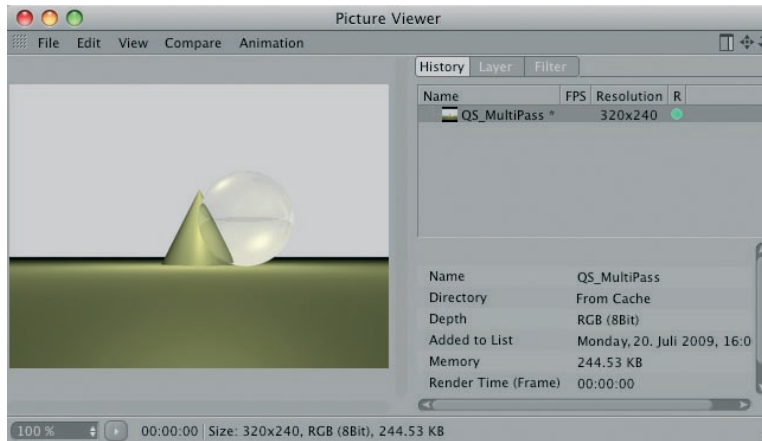
Select the sphere and the cone in the Object Manager. Select the command (Render / Render Active Objects).



Only the selected objects will be rendered.

Rendering the editor view gives us a quick overview of the scene but it does not offer the possibility to process this image further, to save it to the hard drive, for example. What good is the best rendering if you can't save the images it generates? Of course there is a command with which you can do this.

Select "Render / Render to Picture Viewer" or press "Shift+R". The picture viewer will open in a separate window in which the scene will be rendered. When the image has been rendered select "File / Save Picture As". A further window will open. Confirm the location with "OK".



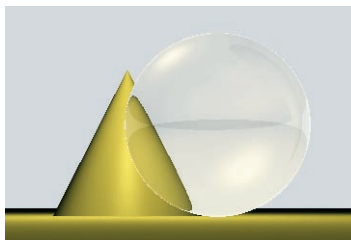
Now you can save any image to your hard drive so you can edit them in an image editing program or just send them to your grandma via email if you want. Rendering to the Picture Viewer has the additional advantage that you can continue working on your scene if the image should take a while to render. You have probably noticed that the image which is rendered to the picture viewer is very small. 320 x 240 pixels to be exact. Why this size? And what should you do if you need a larger format? The time has come to make use of the Render Settings.

Close the picture viewer and open the Render Settings (Render / Render Settings).



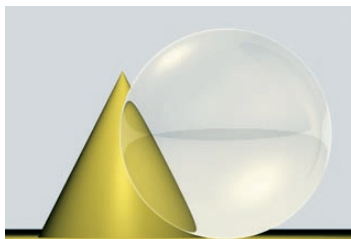
You use the Render Settings to determine what our final image will look like. Everything from size, quality, single image or animation can be set here. Render the image again in the editor window and take a closer look at the result. You can see the cone's edge behind the sphere. It looks a little pixelated. You can see a similar effect along the edge of the sphere. This is called "anti-aliasing". This term refers to how smoothly an edge has been rendered.

Set Anti-aliasing to “None” in the “Render Settings”.



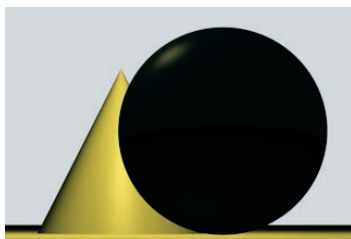
The effect is much worse without anti-aliasing. You can plainly see pixelation along the left edge of the cone now as well.

Set anti-aliasing to “Best” and render the scene again. All edges have been rendered sharp as a knife.



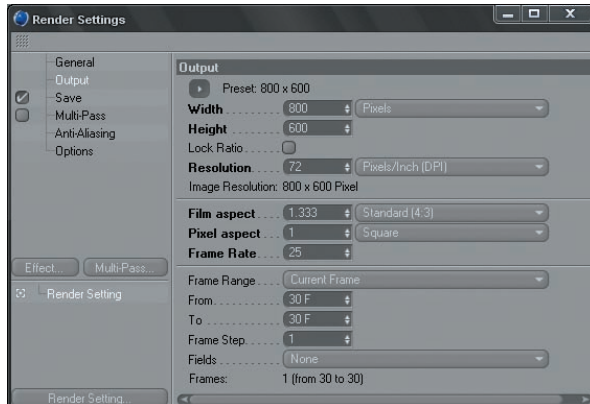
To quickly check the scene you can leave the anti-aliasing set to “None” or “Geometry”. “None” renders the edges without anti-aliasing and very quickly. “Geometry” renders the image with sufficient smoothing and offers a good compromise between quality and speed. You can select the best quality when you render the final image. The “Filter” menu lets you select the type of anti-aliasing filter.

The Render Settings “Transparency”, “Reflection” and “Shadow” can be defined according to how they are needed. Remove the check mark next to the “Transparency” Function and render the scene. You will see that the sphere is no longer transparent.

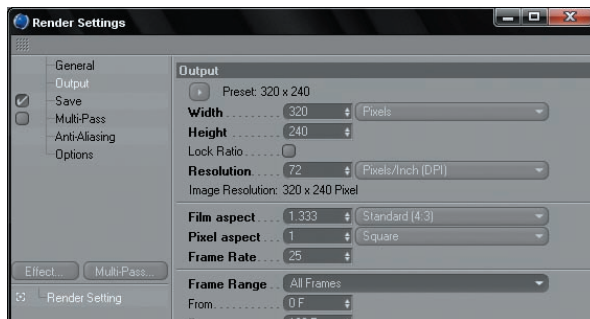


Re-activate the “Transparency” function and switch to “Output” in the Render Settings.

This is where we will find out why the image in the picture viewer is being rendered so small. “Resolution” lets you choose from a wide variety of image resolutions. You can also simply enter the desired image size. Slide the time slider back and forth until you find a frame you would like to render. Set the size to “800 x 600” and render the image in the picture viewer.

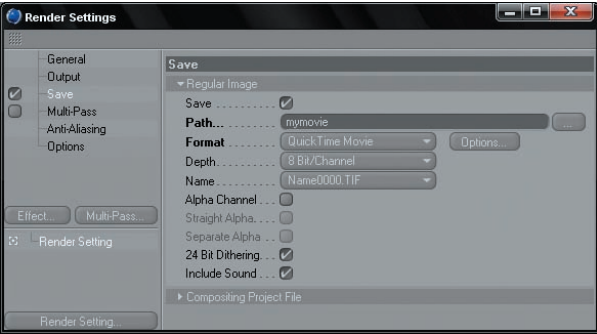


Now you can see a lot more of the scene. We’ve done enough with single images and want to move on to an animated scene. We want to set everything in motion. Set the render size back to “320 x 240” and “Frame” to “All Frames”.



CINEMA 4D will now render all of the scene’s pictures. The scene’s length is determined in the preferences (CINEMA 4D main menu: Edit / Preferences) menu. Our scene runs from frames 0 to 90. We just have to determine which file format our images will have and where they will be saved.

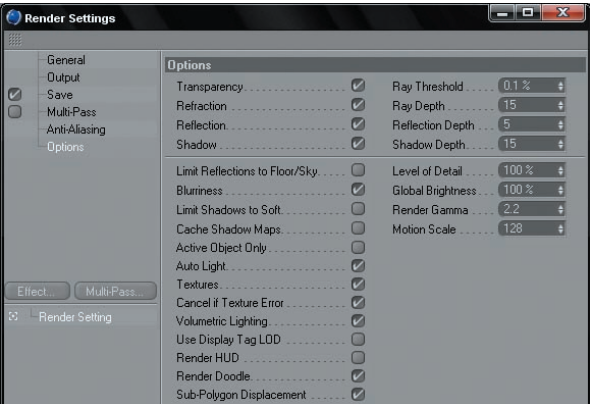
Switch to “Save” in the Render Settings, click on the button next to “Path” and choose a name and a place to save your film. Now choose a format – you can use “Quicktime” or “AVI”.



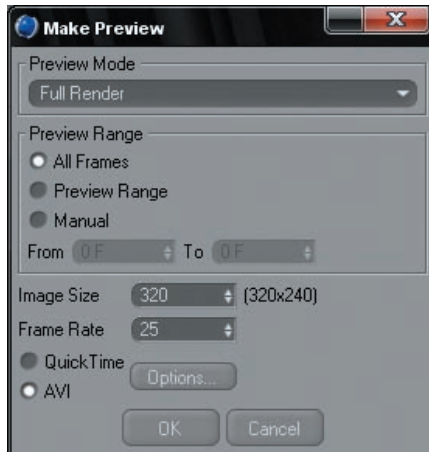
You can also choose single image formats such as TIFF, for example. CINEMA 4D will then save 90 images for this particular animation. You can then create a film from these single images using editing software. Some formats even allow you to save an alpha channel with the image. Alpha channels help you to cut out objects in the image in an image editing program, letting you change the background, for instance.

Select "Render / Render in Picture Viewer" from CINEMA 4D's main menu and watch CINEMA 4D work.

In the "Options" menu of the Render Settings you also will find further settings that influence your rendering. Here you can turn textures off, generate a protocol as a text file and regulate the level of detail.



If all you need is a quick preview of your animation you can save yourself the trouble of always opening and changing the Render Settings by selecting the preview function (Render / Make a Preview). The settings used here are kept to a minimum.



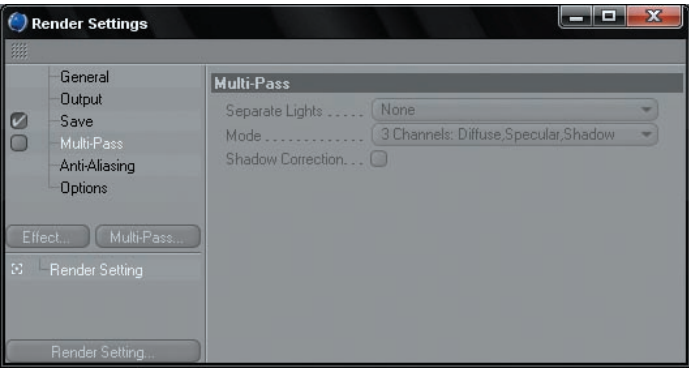
Of course the “Output” and “Save” settings depend on the requirements of your scene. If you render a single image that will be printed with a resolution of 300dpi on a 8.5x11 size page you should render the image with a resolution of at least 2550 x 3300. If you want to print the image in a picture size of 3x5, a render resolution of 1000 x 1500 will be more than enough. By the way, there are many services that will print your digital images. Maybe you can send us your first CINEMA 4D work of art as a Holiday card!

Animation is a different story. The frame rate, which is also editable in the “Output” menu of the Render Settings, plays an important role in animation. The frame rate is the speed at which the animation plays. A frame rate of 25 means that 25 images per second will be played. If you produce an animation for the European market you will have to adhere to the PAL standard which uses an output size of 768 x 576 pixels and a frame rate of 25. If you produce a film the frame rate must be set to 24 and a much higher resolution that for television.

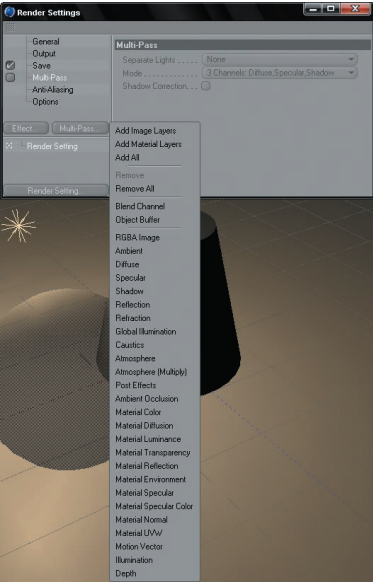
10. Quick Tutorial – Multi-Pass Rendering

As you have read in the previous chapter rendering can take up a lot of time. The last thing you want to have to do is render a long animation a second time because you accidentally set a wrong property. Let’s say you look at your film again the next day and realize the specular light on the sphere was set much too bright. Or worse, you’re a 3D professional and a client is looking over your shoulder telling you he would rather have the pink panther dove blue! Now you have to create a new texture for the character and render the entire animation again – unless you had secretly activated Multi-Pass rendering. With the help of Multi-Pass rendering you can place 3D objects in front of a real background (keyword: compositing) or you can use filters in post-production to make the images more attractive. In this chapter we will show you how to use Multi-Pass rendering so we can take away the hassle of clients who want you to make unexpected changes to a project ASAP.

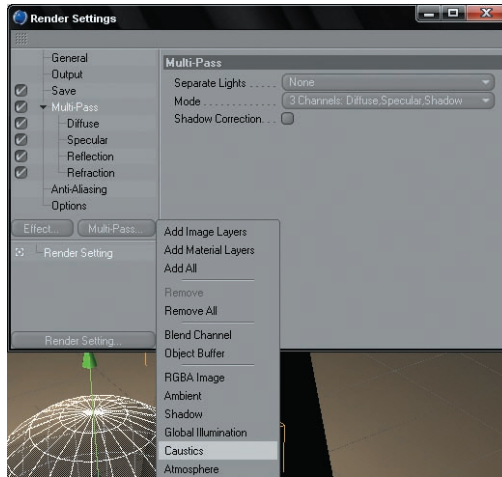
Open the file “QS_MultiPass.c4d”. The time slider is set to frame 15. We want to render this frame and subsequently make changes to it using an image editing software. Open the Render Settings (Render / Render Settings) and select Multi-Pass.



Multi-Pass rendering means that not only the actual image will be saved but also the individual channels (passes) which comprise such an image. A channel can, for example, contain an image's environment light, specular light, transparency or even single objects. This information is output as a separate image. These images can be edited individually using an image editing software and brought together to a single image after editing. To activate the Multi-Pass function, use the check box at the left of the function's name. Further to the bottom is a button named "Multi-Pass..." which will, when clicked, offer numerous Multi-Pass channels that can be added for rendering.

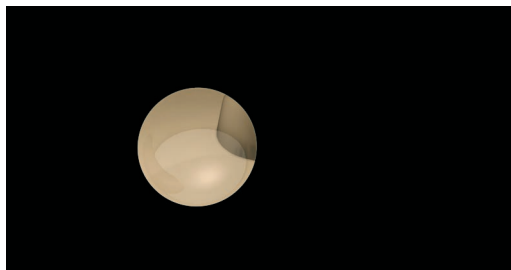


On the “Save” page you will find a set of parameters for saving Multi-Pass files. Set the format to “JPEG” and click on the “Path” button. Find a place to save the image on your hard drive and name the image “Multi-Pass_Scene”. Click on the “Multi-Pass...” button and add the channels “Diffuse”, “Reflection”, “Specular” and “Refraction” and render the scene using the command “Render / Render To Picture Viewer”.



You will not notice anything special when the image is rendered. When you look into the “channel” menu of the image viewer you will see that you can call up each of the channels you just rendered individually. If you now look on your hard drive where you saved the image you will see four files. Since “Layer Name as Suffix” was activated in the Multi-Pass window each image was given a name according to its channel: “Diffuse” for the diffuse lighting, “refl” for the reflection, “specular” for the specular light and “trans” for the transparency / refraction, each with the frame number.

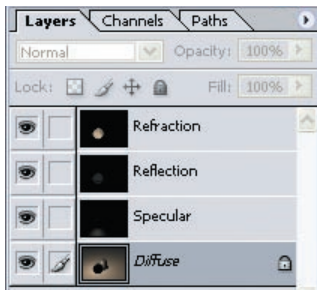
If you take a look at each image you will see that only the information from the individual channel is contained in the respective image. In the “trans” image only the sphere can be seen since it is the only transparent object in the scene.



But how can we use these images to manipulate our scene after it's been rendered? By using an image editing software such as Photoshop. Photoshop is a very popular program which has established itself as *the* standard image editing software. If you don't own a version of Photoshop you can download a demo version from their web site. Photoshop can split an image into several layers. These layers can be mixed in various ways but it would be a lot of work if we had to open each of our images in Photoshop and mix them manually. CINEMA 4D makes it possible for you to integrate all layers into one file. This function does not work for every file format, such as JPEG for example. Since we want to work with Photoshop it makes sense that we use Photoshop's own format "PSD" (Photoshop Document).

Set the format to "Photoshop (PSD)" in the "Save" window's "Format" parameter. If you have Photoshop 7.0 or newer, activate the check box under "Options" at the bottom right of the window. Render the image.

On your hard drive you will now have a file named "Multi-Pass_scene0015.psd". Open the file in Photoshop. The image you just opened looks like the rendered image in CINEMA 4D. Take a look at Photoshop's layer palette and you will see our channels, properly named and linked with each other.



The great thing is that we can change the layers in Photoshop and at the same time the channels of our rendering. Remember the specular light we mentioned at the beginning of the chapter? We think it can be a little brighter.

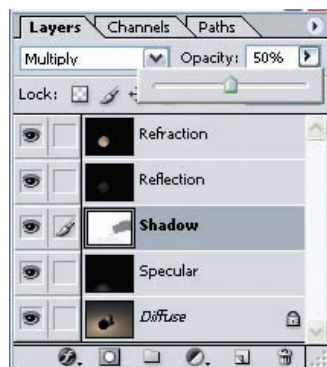
Select the "Specular" layer in Photoshop and call up the "Levels" function (Image / Adjustments / Levels).

Now you've already made the specular light brighter without having to render the image again.



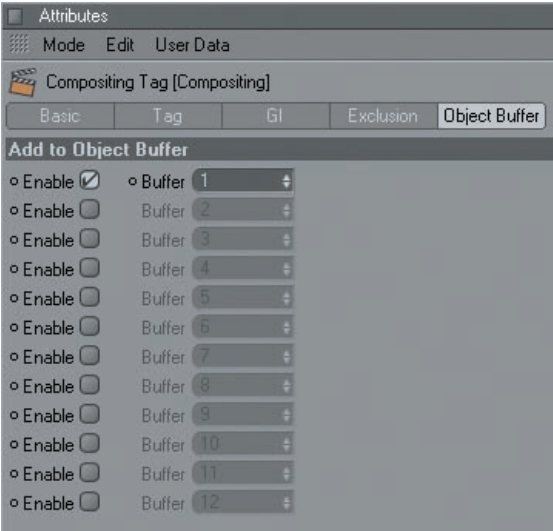
You may have noticed that there was a shadow in the original scene that is not visible in the Photoshop file. This is because the shadow has its own channel and we simply did not activate it in the Multi-Pass rendering menu.

Add this channel in CINEMA 4D, render the image and open it in Photoshop. Now the shadow has its own Photoshop layer. The shadow is very dark and the right edge of the cone can barely be seen. We want to change that now. Select the shadow layer in Photoshop and set its “Opacity” to 50%.

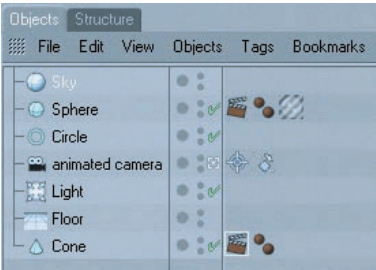


The image already looks brighter. And what should we do with the client who wants his pink panther dove blue? Aside from the fact that our scene contains neither the color pink nor a panther, what should we do when we want to change the color of the cone only? How do we change the color of a single object in the scene? For this we have to prepare a little in CINEMA 4D first.

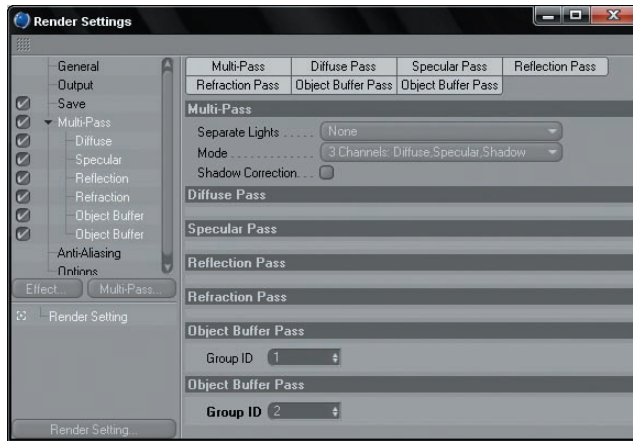
Assign a compositing tag to the sphere in CINEMA 4D's Object Manager under "Tags / CINEMA 4D tags / compositing". Switch to the "Object Buffer" menu of the Attribute Manager and activate "Buffer 1".



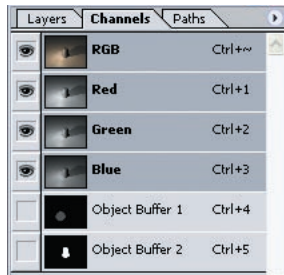
The compositing tag makes it possible to assign an image buffer to one or more objects in the scene. Assign a compositing tag to the cone as well. Activate the second buffer.



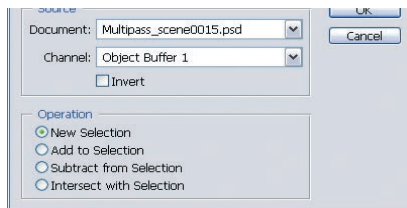
The sphere and cone are now in two separate image buffers, 1 and 2. Now we have to activate these buffers in the Render Settings. Add 2 “Object Buffers” by selecting them from the “Multi-Pass...” menu. For the first, leave the “ID” set to “1” and set the second to “2”. Render the scene and open the new file in Photoshop.



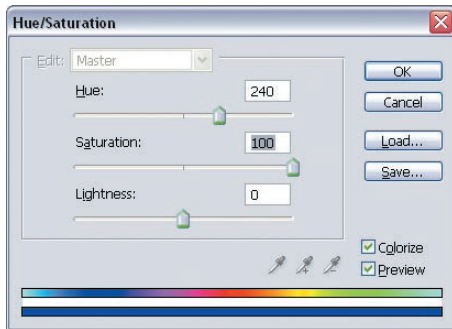
At first glance nothing has changed in the Photoshop file. There are still only five layers. Where are the new objects? They can be found in the channel palette. An “Alpha Channel” was created for each object. An alpha channel is a gray scale image used to cut out objects in the image.



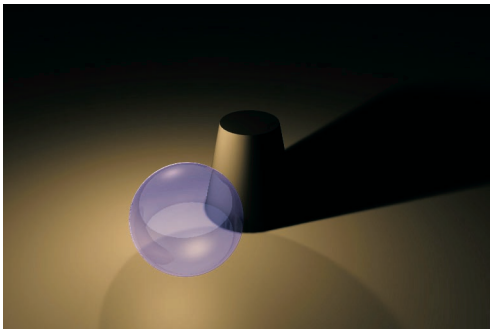
Select the “Refraction” layer in Photoshop. Choose select / load selection. Set “Channel” to “Object Buffer 1”.



We just loaded the buffer channel as the selection. Various image editing options will now be limited to the area within this selection – the sphere. Call up “image / adjustments / hue / saturation. Click on “Colorize” and set the “Hue” to 240 and the saturation to 100.



Only the sphere will be colored. Since it consists mainly of refractions, we chose the respective layer in advance.

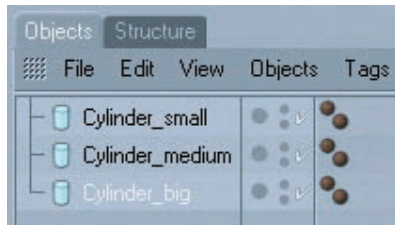


Multi-Pass really comes to shine when it's used with animated scenes. What Photoshop does for single images, compositing programs do for animations. After Effects and Combustion are two of these programs. Editing software such as Final Cut lets you edit image layers as well. CINEMA 4D works with all of these programs, partly over special export plugins which offer an extensive number of settings. You can change a color in a 15-minute animation in seconds.

11. Quick Tutorial – XPresso

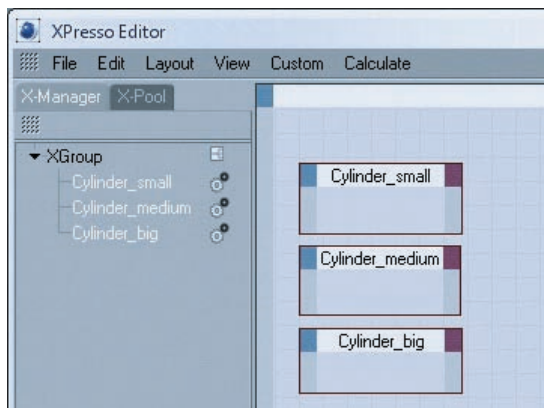
Expressions let you set dependencies for object properties. In other words, you can use rules to automate a scene: "When object A performs action B object C should perform action D." MAXON created XPresso so you don't have to enter these expressions manually. XPresso is a graphic interface which puts an end to the cumbersome act of typing code. XPresso lets you create expressions using drag & drop which is simple (and can even be fun) and offers enough functionality to be able to solve just about any problem. We will now take a look at what this problem solving looks like. In this tutorial we will build a kind of telescopic arm.

Open a new scene. Create three cylinders (objects / primitives / cylinder). Rename them into "cylinder_small", "cylinder_medium" and "cylinder_big". In the Attribute Manager set the radius of "cylinder_medium" to 45, the radius of "cylinder_small" should be 40.



We now have three cylinders with different radii in the scene. Our goal will be a telescopic effect by pulling the smallest cylinder up while keeping the medium cylinder dead center as it moves with the smallest cylinder. This is a case for XPresso!

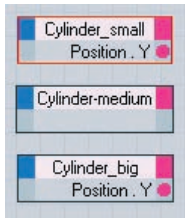
Apply a new XPresso tag. The quickest way to do this is to click on one of the cylinders in the Object Manager with your right mouse button. Select CINEMA 4D Tags / XPresso. By the way, it doesn't matter which object you apply the tag to. The XPresso editor will open. Drag all three cylinders from the Object Manager onto the empty surface of the XPresso editor. The objects will be transformed into little boxes, so-called "Nodes".



Nodes are XPresso's cornerstones. They represent objects or functions. Nodes have an input and an output which allow them to exchange information with other nodes. You can view a node's complete input by clicking on the little blue box in the top left corner of the node. A node's output can be viewed by clicking on the red box. Take a look at the exits of the cylinder's nodes. Some of them will surely be familiar to you (Attribute Manager).

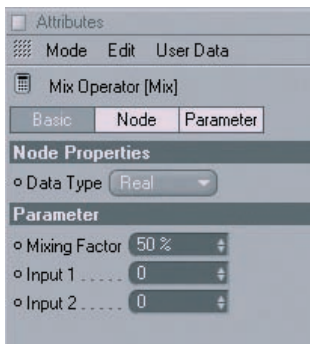
Think about what it is we want to happen. We want to pull the small cylinder up, i.e.; change a Y-value. This movement should in turn influence the Y-value of the medium cylinder which should stay exactly in the center of the Y-positions of the big and small cylinders. We can say that we want to set the medium cylinder to an average value of the other two cylinders.

Click on "Cylinder_big's" red corner and select Coordinates / Position / Position Y. Do the same for "Cylinder_small".

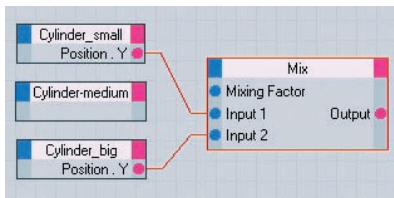


Both cylinders can now pass on their respective Y-position. We will use a special node to calculate the average between these two points.

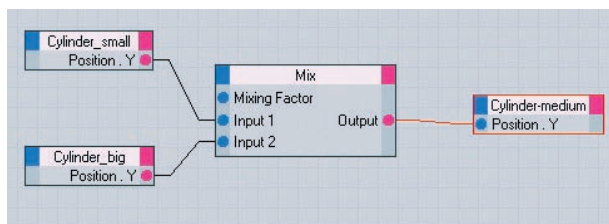
Switch to the X-Pool tab and drag a “Mix Operator” node from System Operators / XPresso / Calculate / Mix onto the surface where the other nodes are (small tab at the top left next to the X-Manager). Set the “Mix” node’s “Mixing Factor” to 50% in the Attribute Manager.



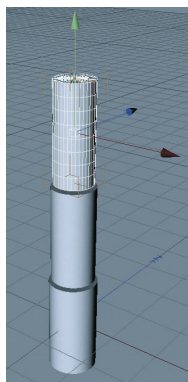
Connect the “Cylinder_small” “Position.Y” output to the “Mix” node’s input 1 and the “Cylinder_big” exit to the “Mix” node’s input 2.



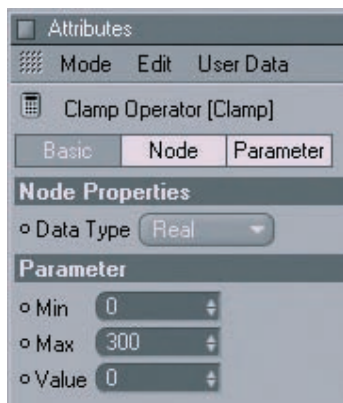
Both Y-positions will now be transmitted to the “Mix” node which will now calculate their average of their respective values. The 50% setting makes sure the values will be mixed evenly. The result will be the Y-position of the medium cylinder. Click on the blue corner of the “Cylinder_medium” node and open an input for the Y-position (Coordinates / Position / Position.Y). Connect the “Mix” node’s output with the input of “Cylinder_medium” node.



Now gaze in amazement at the result in the editor window. Move the small cylinder along its green Y-axis. The center cylinder aligns itself automatically.

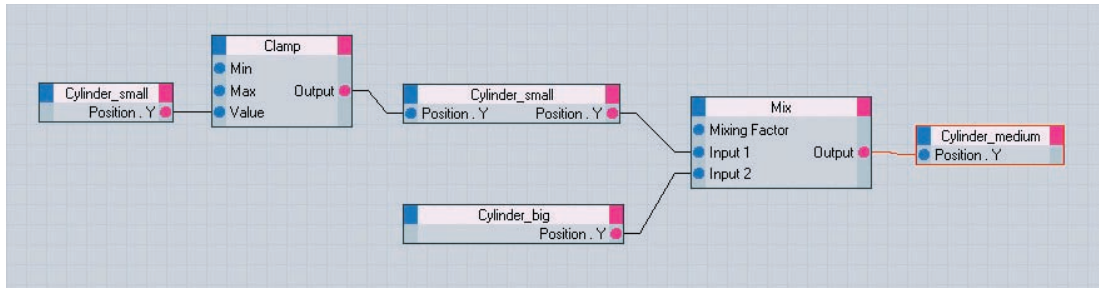


We still have one problem: you can still drag the cylinder so far that our telescope comes apart. It would be nice if we could restrict the movement a little. Needless to say XPresso also has a solution for this! Go back to X-Pool and drag a “Clamp” node into the XPresso editor. Set its “Max” value to 300 in the Attribute Manager.



The “Clamp” node restricts various properties that are transferred to its “Value” input to a range between 0 and 300. Now we just have to connect the node to the rest of the setup.

Click on the blue corner of the "Cylinder_small" node and open an input for the Y-position (Coordinates / Position / Position.Y). Drag "Cylinder_small" from the Object Manager into the XPresso editor again and open an output for the Y-position. Connect this output to the "Value" input of the "Clamp" node. Connect the output of the "Clamp" node to the "Position.Y" input of the "old" "Cylinder_small" node.



Now the cylinders can only be moved a limited distance in the Y-direction, just like a real telescope.

12. Tips & Tricks



© Luxigon, www.luxigon.com

CINEMA 4D has been able to build a large community of users around it who are more than happy to help newcomers in any way possible, be it through the use of home-made tutorials, directly in one of the many forums or by offering free models, plugins, expressions or textures.

One of the main sources of information, of course, is the MAXON web site www.MAXON.net.

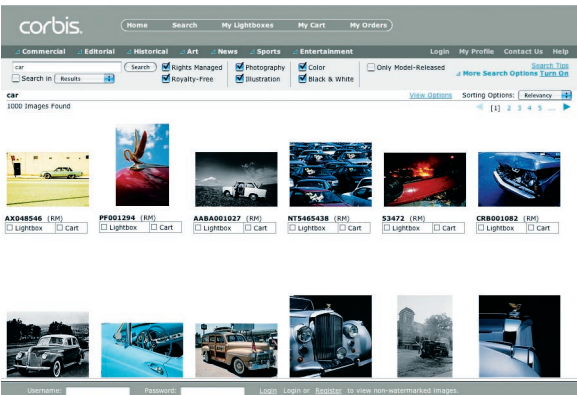
Here you will find news, interesting projects that were done with CINEMA 4D, an extensive CINEMA 4D link library and even a form for questions for our support department. You can even get information about third party seminars and training.

A large selection of books is also available for those who would like to expand their CINEMA 4D library. There are books from the basics to specialization such as character animation. Just enter the keyword CINEMA 4D into the search engine of your favorite **online book store**.

Flipping through books which don't have anything to do with computer graphics can also be helpful. 3D is a complex field in which many of the classic arts and techniques are combined. Books about photography, lighting, direction, acting, image creation and painting should be part of every serious 3D artist's library. In addition, you will find a wealth of information on all of these topics online.

Since the internet is always changing, doing a search for "CINEMA 4D" in various search engines would be a good idea.

The Internet is a good source for finding models. Through its special image search function you can find photos or drawings of practically any object.



Even mail order catalogues can be a great source of information on how an object is supposed to look. Textures are all over the internet as well. Taking your camera and photographing textures yourself is even better. Inspiration is everywhere. You can build your own texture archives in no time.

Try to get away from the technical point of view. Learning a software is only a matter of time. A good 3D artist has the ability to use software as a tool that helps him realize his ideas. The real creativity lies with the idea, not the software. So when you create your next scene don't worry so much about creating the perfect object. Concentrate rather on how you can make a harmonious composition with this object with a fitting theme and proper lighting. Also, think about the message you are trying to send to the viewer. The same goes for animators. A technically perfect animation is a great achievement but it will put your audience to sleep if the concept is bad. It's not so bad, on the other hand, if your animation is a little bumpy and imperfect but your story touches the viewer.

We hope this manual has helped you to master the technical part of 3D graphics. What you do with what you've learned is in your hands.

Welcome to BodyPaint 3D

This is the BodyPaint 3D tutorial. In this tutorial we will explain the most important functions in order to give you a running start in the world of “body painting”. Even if BodyPaint 3D appears to be difficult at first, you will soon notice how intuitive BodyPaint 3D really is. In this tutorial we have also put an emphasis on a fast learning curve and a high degree of user friendliness for this module. Let’s start with its structure.



© Kornel Ravadits

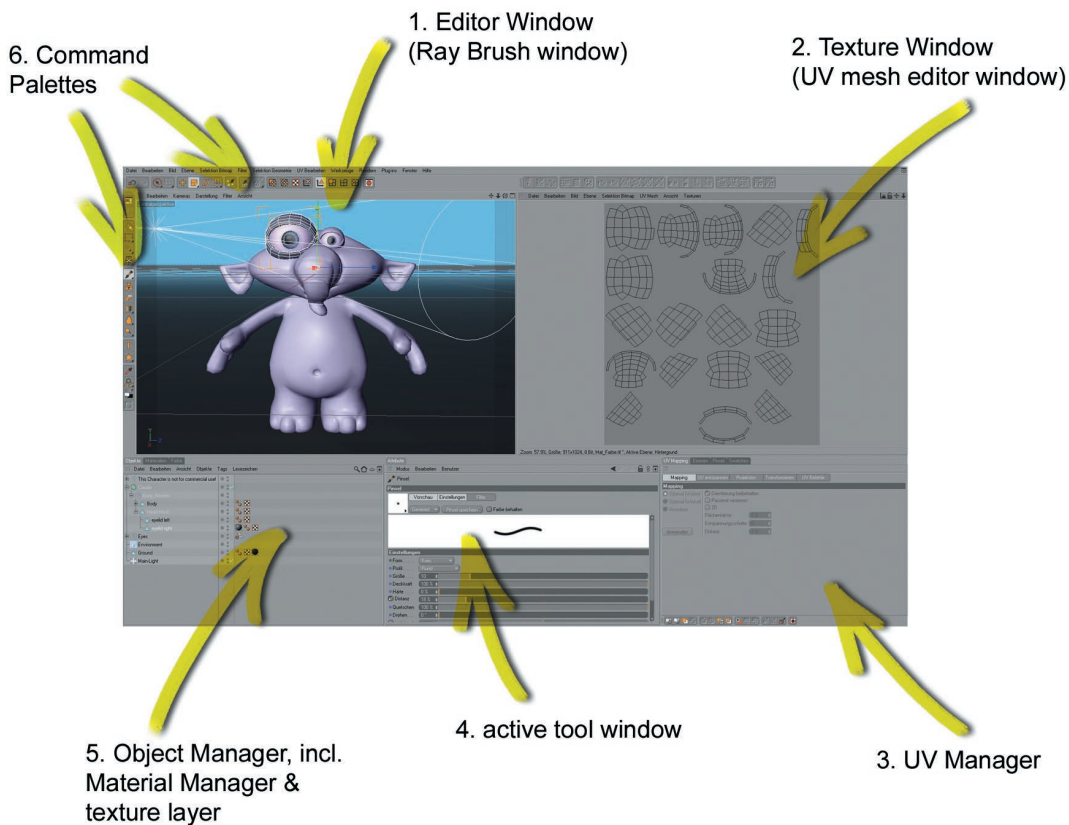
1. Introduction

BodyPaint 3D will revolutionize the way you work with textures in such a way you will wonder how you ever got along without it! With this module you can paint your models as they are: in 3D. This is what BodyPaint 3D, the revolutionary way to texture objects, is all about. In addition, BodyPaint 3D lets you paint in several texture channels at once, and thanks to RayBrush even directly on the rendered image itself. Projection Painting is a tool we have integrated that makes it possible to paint on complex objects without distortion.

Using the UV-tools you can relax and stretch your UV-mesh, no matter how complex it is. Put simply, a UV-mesh is a second impression of a polygon mesh that projects the texture onto a polygon object. The days of 2D texturing are over and you can finally concentrate on what's important in texturing: creativity. Everything that took up so much time with 2D texturing is now done by BodyPaint 3D and you can deliver your projects faster. Let's move to the user interface.

2. General Information / Interface

First take a look at the screenshot on the next page.



Here you see one of the two standard layouts: “BP UV Edit”. The second layout (“BP 3D Paint”) is set up in a similar fashion, only without the UV-mesh editor window which gives you more room in the editor window to paint.

1. Editor Window (RayBrush Window)

Here you can see the object you will be painting. You can rotate, move and zoom the window as needed. The RayBrush mode lets you paint directly onto the object in the rendered version of the view. This gives you control over the amount of color applied and can see right away how a new color looks on the object.

2. Texture Window (UV-mesh Editor Window)

This is where you edit your UV-mesh. You can relax and restore your UV-mesh. If you use the UV-Manager’s UV-tools you can watch how the texture relaxes. You can also watch the color application process in this window, which will then be visible in the editor window right away.

3. UV-Manager

The UV-Manager lets you restore the UV-mesh using an algorithm. It recognizes layered polygons and attempts to relax the UV-mesh for optimal placement over the entire surface and, if necessary, new placement. All remaining “relaxation” can be adjusted manually.

4. “Active Tool” Window

Different tabs display different brush types and their respective attributes as well as the UV Manager’s UV tools.

5. Material Manager and Texture Layers

We’re sure we don’t have to say much about the Object Manager. It’s the same manager as in the CINEMA 4D main program and lets you select the object to be edited or change its position in the hierarchy.<

This is the CINEMA 4D Material Manager with expanded functionality. This is where you will find your textures with their respective layers. If needed, you can paint in several layers at once (for example color and Bump channels). To do this simply select the texture to be painted and the respective layer and start painting.

6. Command Palette

The command palette contains the Paint Wizard, the Projection Painting and many other tools (that you’re probably used to using with 2D paint programs). The BodyPaint 3D Wizard eliminates the need to manually create a texture including the UV-mesh. It also calculates the texture size and channels. Without these bothersome preparations you can begin painting right away.

3. Sample Images

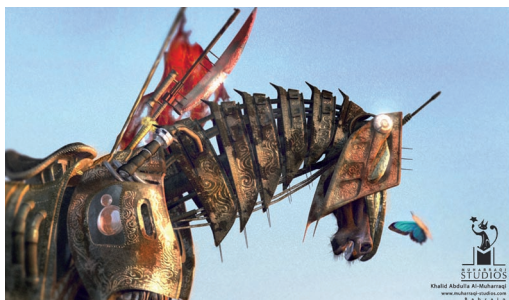
We've reached a part of the tutorial for which words are not necessarily needed. Simply take a look at the following images.



© Patrick Eischen, www.patrickeischen.com



© James Ku



© Khalid Abdulla Al-Muharraqi
www.muhamarraqi-studios.com, Kingdom of Bahrain

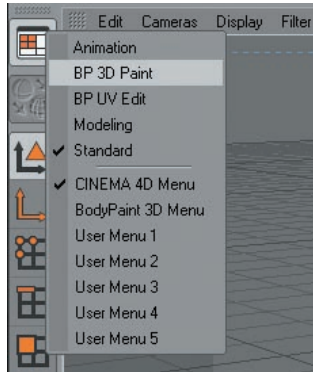


© www.bediff.com

4. Quick Tutorial – The Paint Wizard

The BodyPaint 3D Paint Wizard takes a lot of preparatory work off your hands and lets you begin texturing / painting in seconds. Before we actually start painting the object we would like to show you how quickly you can start painting, just in a few steps.

Create a cone primitive (objects / primitive / cone). Switch to the predefined standard layout “BP 3D Paint”.



Click on the “Paint Setup Wizard” icon



and select the “Next” button twice, the “Finish” button once and to exit the BodyPaint 3D wizard click on “Close”. At the left of the command palette select the “Brush Tool for Painting Textures”



and drag the cursor over the cone while pressing the left mouse button. Voila!



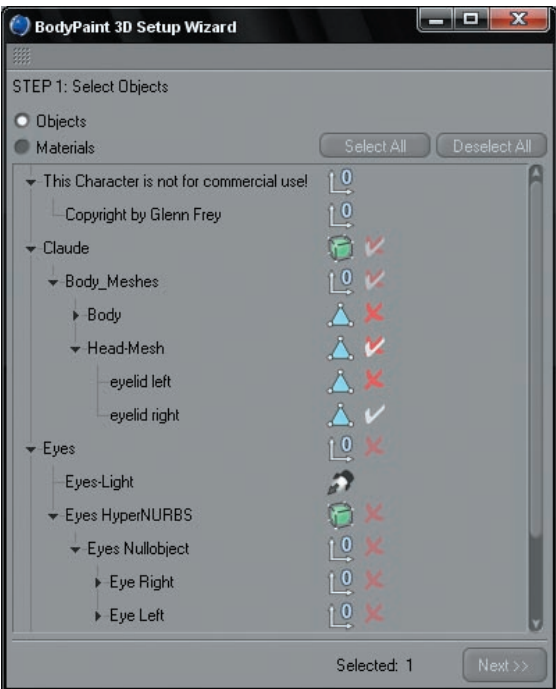
Aside from the fact that you will never have a need for a cone with a white mark on it, this simply demonstrates how BodyPaint 3D works.

Now we'll get to the heart of this tutorial. Open the file "QS_BP3D_Startl.c4d". Say hello to Claude, our guinea pig for the day. In the course of this tutorial we will alter the color of his right eyelid a little and apply a bump layer in elephant-look to his skin.

Select the predefined standard layout "BP UV Edit" at the top and to the left of CINEMA 4D's main editor window. Click on the BodyPaint 3D Wizard Icon so we can make the necessary preparations to the texture.



Click on "Deselect All" in the window you just opened and apply a white check mark to the "eyelid right" object only.

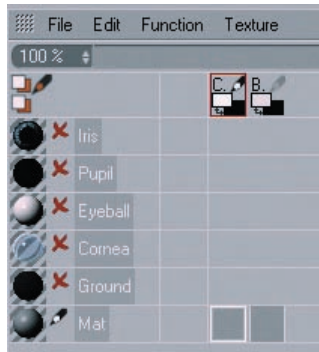


We have just determined that a texture should be created only for the right eyelid object. Click on "Next". Leave the settings in the next window the way they are. The selection "Single Material Mode" would create a texture for each object individually. If the box is not checked all objects will share one texture surface. Click on "Next" again. In the next window check the bump channel. The color channel is selected by default. You can double click the little gray boxes next to each texture channel and assign each channel a base color.

Since Claude likes elephant gray we will leave the boxes the way they are. Leave the rest of the settings the way they are and click on “Finish”, then on “Close” in the next window. The basic textures have been created and we can start painting. If you have experience with earlier texturing methods and the time it took to even get started BodyPaint 3D will seem like a blessing to you. BodyPaint 3D saves you a lot of time. Now let’s move to the second part of the tutorial: the UV-meshes and the first brush stroke.

5. Quick Tutorial – First Painting Lesson

At the bottom left of the Material Manager (and other Managers) you will find the texture we just created, right next to “Mat” in the “Material” tab.

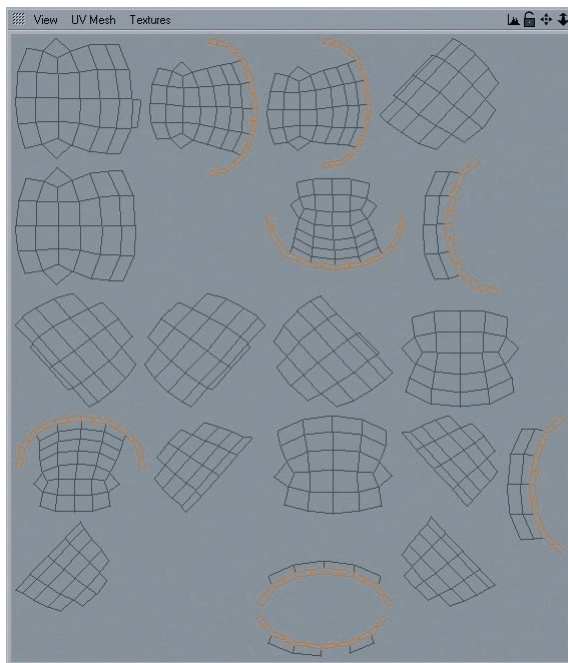


This is the default name for a new texture. Of course you can rename the texture if you like. The first material is the color layer and the second is the bump layer (at the top of the window you will see the abbreviations which refer to these layers – “C” for color and “B” for bump).

Now Select the “Use UV Polygon Edit Tool” symbol.

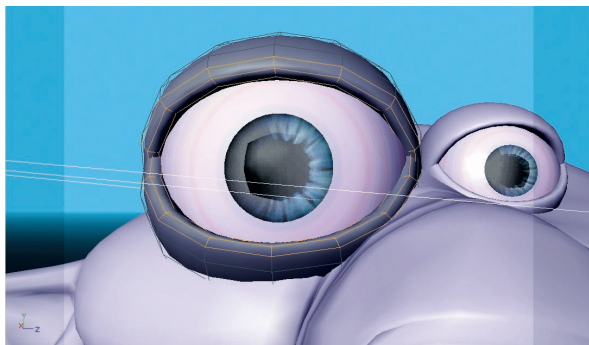


Once you have selected the corresponding texture in the color channel the UV-mesh should become visible in the texture window at the upper right. If the mesh is not visible, activate it by clicking on “UV Mesh / Show UV Mesh” in the texture window menu. Luck is on our side! The UV-mesh looks good. The only thing that bothers us is the fact that the edges of the eyelids are too small (highlighted in orange in the next image!).

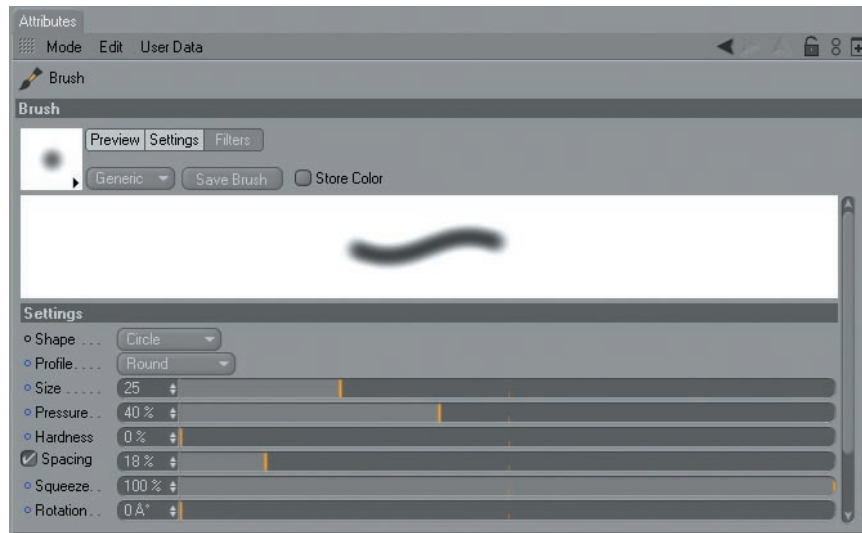


The individual UV-mesh polygons of these eyelid edges take up less texture area than the rest of the polygons. That's why a texture placed into the bump channel appears larger in these places (photograph of elephant skin, for example). We can do without this, though, since we are painting our own skin structures onto the surfaces and not using an existing texture. We can counter any distortion we encounter when painting manually by using "Projection Painting". The stroke will maintain its width no matter how the polygon is spread over the mesh.

Move and zoom the editor window view until Claud's right eyelid fills the view.



Select the brush on the command palette on the left, Now select “Brush Tool for Painting Textures” for applying the color. Set the size to 25 and the hardness to 40 in the brush’s Attribute Manager (“Active Tool” window).



and select a pink color in the Manager directly above the “Color” tab.

If necessary, increase the HyperNURBS subdivision. Activate the “Render Active View for RayBrush Painting” in the active view in the “Render” menu (BodyPaint 3D main menu).

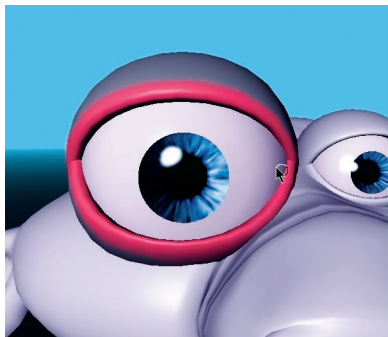


(This will render the view and makes it possible for you to control the color application and the look of the strokes for the final rendering). Activate the “Enable / Disable Projection Painting”



(You already know what this function does) and start painting. Of course BodyPaint 3D supports the use of graphic tablets such as a WACOM Intuous. Painting objects with a pressure sensitive pen on a graphic tablet is much easier than painting with a mouse.

Paint along the edge of the eyelid. The eyelid will probably end up looking like this:



If you move / rotate the figure now or click on the “Apply the Content of Projection Paint Plane” (click and hold on the “Activate/Deactivate Projection Painting” button)



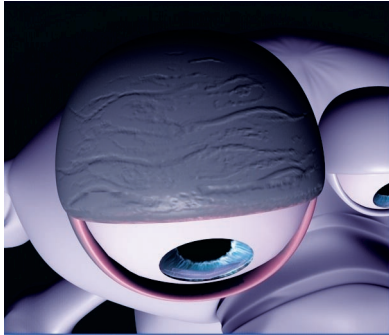
you will see how the color was applied to the texture (you can see the recently applied strokes of color in the window to the right).



You can take the same steps for the bump layer. We will take you one step further, though, in order to be able to explain an important function. We will paint both eyelids at the same time! Select the texture in the color channel of the Material Manager. Now click on the icon with the black/orange pencil at the left of the Material Manager. A light gray background tells you the multi-brush mode is active. Select the pencil icon next to the “B” of the bump channel as well.

You have now told BodyPaint 3D that you want to paint in both layers at the same time. If you like you can switch from the standard “BP UV Edit” layout to the “BP 3D Paint” layout. This gives you more room to work in the editor window.

Rotate the view so you can see the eyelid from the top. Activate the “Render Active View for RayBrush Painting” and the “Enable / Disable Projection Painting” mode and set the brush size to 10 and hardness to 40. Switch to the color layer’s “Color” menu and set the color to a medium gray which will be the base color for our eyelid. Now go to the bump layer’s color preview and set the color to black (both color layers are located at the left of the Object or Material Manager in the “Material” tab under the letters “F” and “B” + pencil symbol). When you paint on the object you will notice that both colors are being applied to the object – the gray base color and the black (to indicate indentations). (If white were the color of the Bump channel it would “raise” the brush stroke instead of indicating indentations). The result could look like the following image.



Load the “QS_BP3D_Final.c4d” file and take a look at it when you have time. Here are some everyday tips with which you can achieve great results quickly and easily.

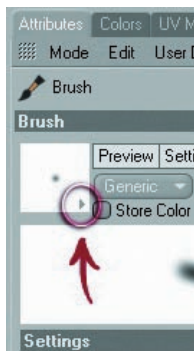
6. Tips & Tricks

A very helpful function can be found in CINEMA 4D’s preferences (Ctrl+E). In the “**BodyPaint**” menu you will find the function “Project On Invisible Parts”. Which, when activated, can make your work a lot easier. Let’s assume you want to color the arm of a figure or sprinkle color on the entire figure. You would have to apply the color with this function deactivated, rotate the arm, apply the color, rotate the arm and, well, you get the idea. When this function is activated you apply the color in the front view and the color is applied to all surfaces lying behind this surface at the same time. Just make sure you don’t apply color to objects you don’t want to color when this function has been activated.

If a texture map does not fit correctly at the point where large and small polygons meet (in the case of low-poly objects that are subordinates of HyperNURBS) set the “Tile UV’s” function, in the respective HyperNURBS Object’s Attribute Manager, from “User” to “Border” or “Edge”. This sends the UV-mesh through the HyperNURBS algorithm and subdivides it to fit the polygon object.

Avoid UV-mesh polygons that meet to a point when applying a “noise texture” to a bump layer. The narrower a 3-sided polygon becomes, the coarser the bump noise channel will be rendered. Of course such a polygon has much less area for the noise structure at its tip than it does at its center which results in a magnification effect of the noise structure.

Try to set up each side of a triangulated polygon as an isosceles. This also goes for “4-point polygons” when they converge into a trapezoid. The more square the polygon the more even the structure will be. It goes without saying that you need different brushes for different texture looks. CINEMA 4D has a wide variety of brush types for you to use. Just select the tab “Attributes” and click on the small arrow on the brush preview



Here you will find all the brushes your heart desires. If you don't find the brush you're looking for we've beat Murphy to the punch and have given you the possibility to create and save your own brushes. Just make the changes you want and click on the “Add Preset Save Brush” button. With this tutorial you have gotten to know how BodyPaint 3D works and you can convince yourself of the advantages painting directly onto objects themselves offers. With only a little practice you can also achieve similar results as you can see on the next image – Claude's new texture outfit.



Here the same rule applies for best results: Try it!

Welcome to Advanced Render (Optional module)

This is the Quickstart Tutorial for the Advanced Render Module. It will show you many of the module's typical applications and covers some theoretical physics behind the effects.



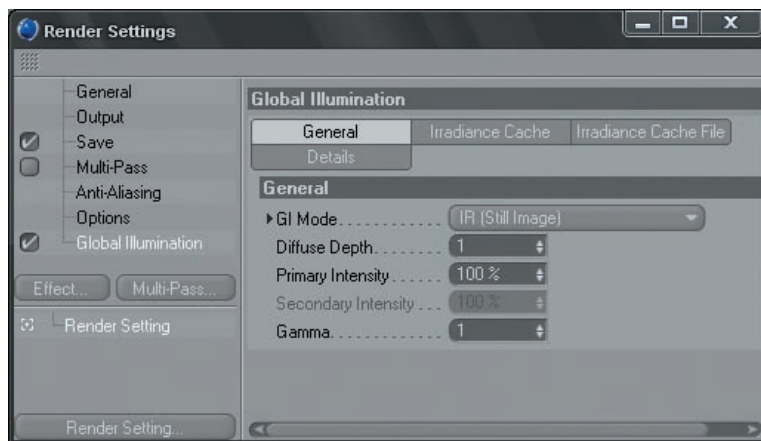
© Dave Davidson, www.max3d.org

1. Introduction

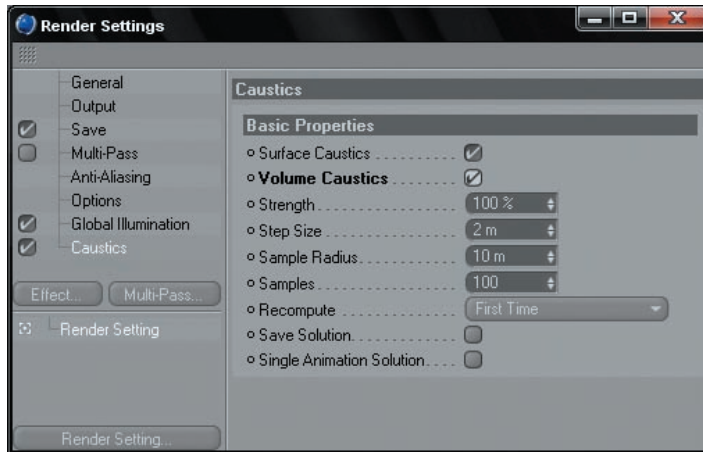
The Advanced Render Module expands the realm of possibilities of CINEMA 4D's renderer with several fascinating functions. Users who want to make photorealistic renderings will find these functions especially interesting. The module integrates itself seamlessly into existing rendering settings which makes learning to use it a simple matter. The module adds the following functions to the standard renderer: Global Illumination (a rendering method for especially realistic lighting), caustics (achieves lighting effects on mirrored and transparent materials), depth of field, specular lighting, glows and sub-poly displacement.

2. General Information / Interface

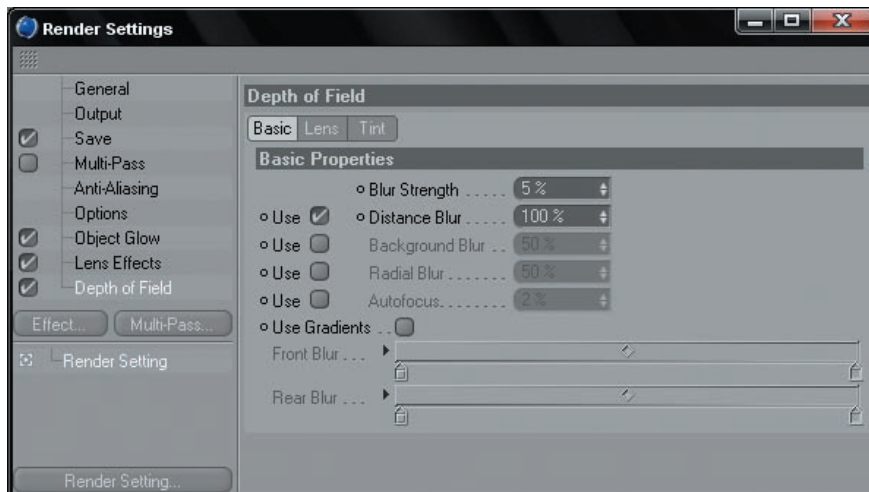
As soon as you have installed the Advanced Render module you will find the "Global Illumination" by clicking on the "Effects..." button in the Render Settings. As soon as you have activated "Global Illumination Rendering", other options will be made active which allow you to create custom settings for radiosity. A material's "illumination" settings are closely associated with this field. These settings let you determine if a particular material should be rendered with Global Illumination. Further settings can be applied to individual objects using the "compositing tag".



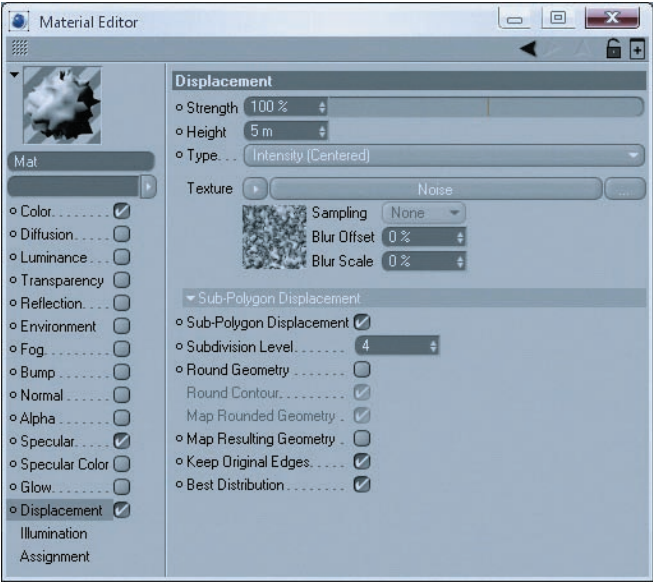
Caustics acts in a similar fashion. The global settings can be found under "render settings". Here you can activate or deactivate surface and volume caustics separately. Options relating to specific materials can be set in the luminance channel. In addition, caustics also offers a third setting. You will find this in the light object. The use of caustics requires at least one light object. Within a light's caustics menu it is possible to determine whether the light source should generate surface or volume caustics.



“Depth of field”, “highlight” and “glow” are post effects. You will find them in the Render Settings under “effects”. Further settings for “depth of field” can be made in each camera’s attribute settings. A post effect will first be calculated after an image has been completely rendered. You can imagine it as a layer which is placed on top of the image after it has been rendered.



The Sub-Polygon Displacement can be activated in a material’s “displacement” channel.

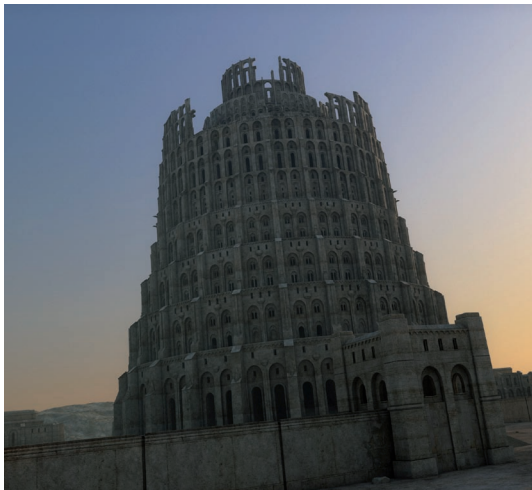


3. Sample Images

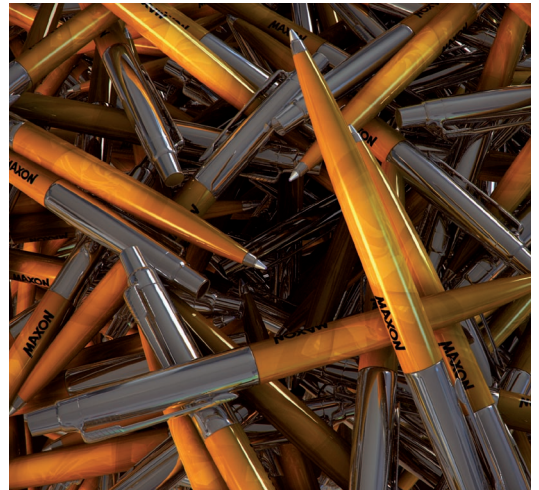
Here you can see what's possible with the Advanced Render functions.



© Marco Weiss, www.black-graphics.de



© www.ericsmitt.com



© Matthew O'Neill, www.3dfluff.com



© James Kaufeldt



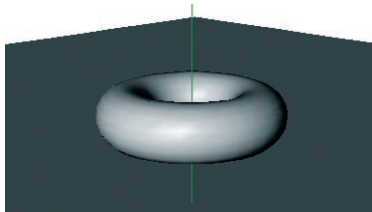
© Olcay Kayihan, olcay@aluplan.com.tr

4. Quick Tutorial – Global Illumination

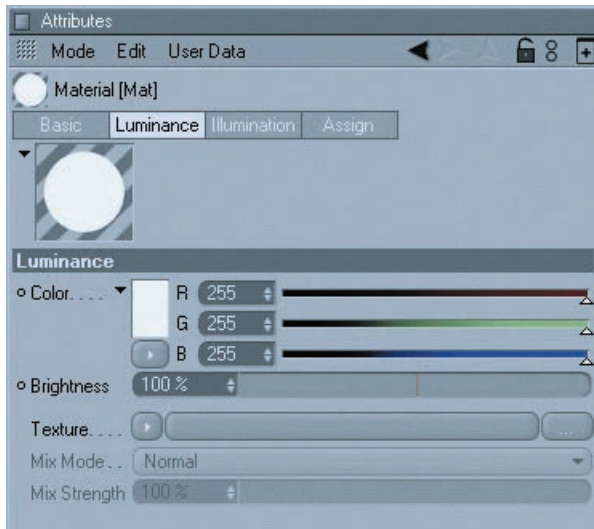
Light as we know it in the “real” world spreads on its own. It is reflected by the objects it hits. This differs depending on an object’s surface characteristics. Imagine a room with a window on one wall. Light is being cast through the window onto the floor of the room. The light doesn’t stop there, but is reflected from the floor onto other objects which, in turn, reflect the light themselves. The room is lit by “diffused” (indirect) light.

The raytracing procedure takes into account diffused light. For example, if only one light is used, everything lying in the shadows will not be visible. Maybe you have already built a virtual room into which a light source is shining through a hole in the wall. The light in the virtual world hits an object, lights it – and that’s it. The light spreads no further. Global Illumination rendering is different. Global Illumination rendering lets every object within the scene act as a light source. As you will see, you can actually light a Global Illumination scene without using a single light!

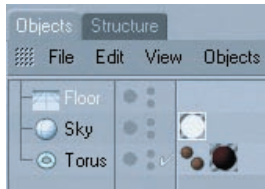
The CINEMA 4D GI rendering engine was completely re-written for Release 11 and is now more powerful than ever! GI rendering has been made much faster and offers even better results thanks to reduced artefacting and flicker-free animation rendering. The GI interface has been redesigned to make it intuitive to use, for novices and professionals alike. Open a new (empty) file. Create a sky object (objects / scene / sky) and a floor object (objects / scene / floor). The sky object encompasses the entire scene like a large sphere. The floor surface is an infinite surface. Create a torus (objects / primitive / torus) and move it to a y-position of 100, slightly above the floor.



Now we will light the scene with diffused light using Global Illumination rendering. We will use our sky object as the light source. Switch to the Material Manager. Create a new material (file / new material). Switch to “basic” in the Attribute Manager. Deactivate “color” and “specular” and activate “luminance”. Drag the material from the Material Manager onto the sky object in the Object Manager.

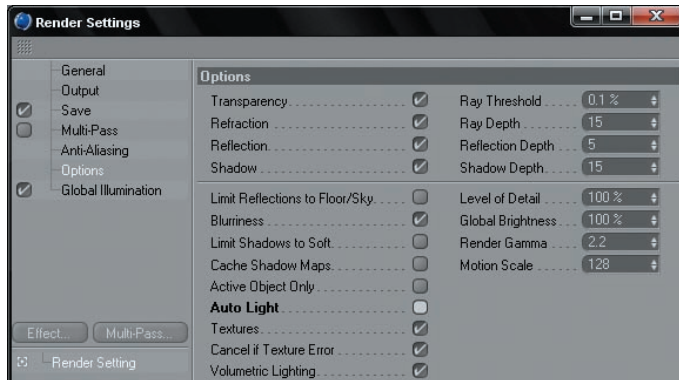


Create another material and give it your favorite color. Drag this color onto the torus.

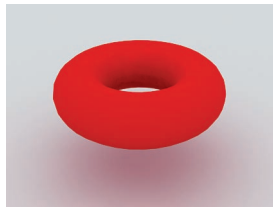


The luminance channel turns the material into a light source. Since the sky object spherically encloses the entire scene. It acts as a huge lamp which lights the torus from all sides. This effect will only be visible when we use Global Illumination as the render mode.

Open the Render Settings (Render / Render Settings). Click on the "Effect..." button and Global Illumination. Switch to the Render Settings' Options menu and make sure that "Auto Light" is deactivated.

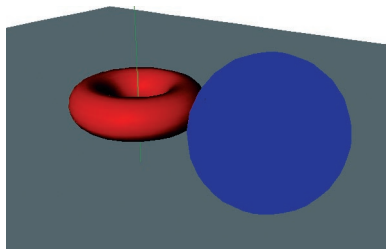


Rotate the scene in the editor so the view is at such an angle as to show only the floor in the background. This speeds up rendering since the rendering will only be done to the “horizon”. Render the scene.



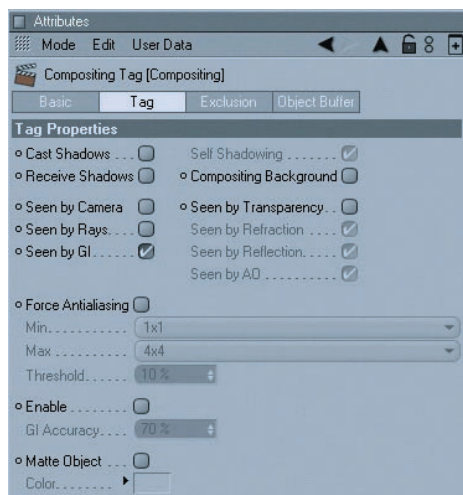
CINEMA 4D will automatically turn on Auto Light in a scene if there are no light objects present. When using Global Illumination, this automatic function is excluded since it would make the scene much too bright.

Create a sphere (objects / primitive / sphere). Move the sphere along the X-axis a little to the right and a Y-position of 100 until it’s next to the Torus. Copy the sky material and switch to the luminance channel of the new material. Use the color sliders to create a bright blue. Drag the material onto the sphere.

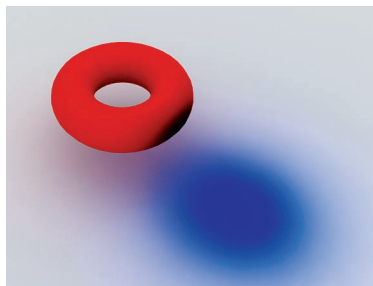


We now want to use the blue sphere as a light. We don’t want the sphere to be visible, but only to emit its blue color. You achieve this by using the “compositing tag”.

Apply a compositing tag to the sphere in the Object Manager (Tags / CINEMA 4D Tags / Compositing). Switch to the Attribute Manager and deactivate all options except “Seen by GI” (Global Illumination). Render the scene.



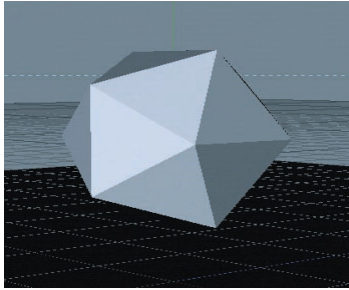
You will see that a blue light is being cast on the torus and the floor. The blue sphere is not rendered because we have made it invisible using the camera's compositing tag.



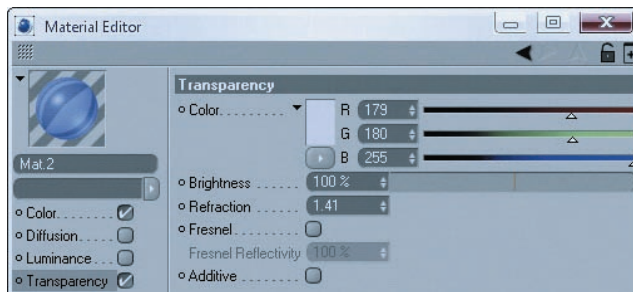
5. Quick Tutorial – Caustics

Have you ever observed the lighting effects at the bottom of a swimming pool? This phenomena is known as caustics – light is cast through a transparent material and is broken upon entry and exit.

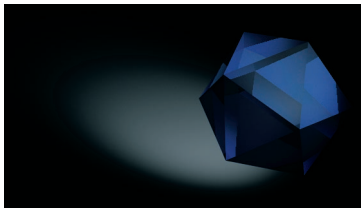
Open another new (empty) scene. Create a floor object (objects / scene / floor). Create a platonic (objects / scene / platonic) and give it a Y-position of 140 using the Attribute Manager.



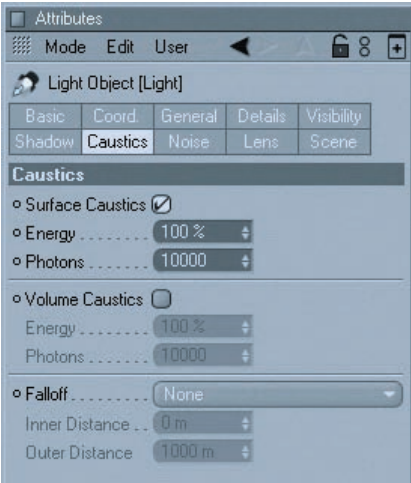
Create a new material in the Material Manager. Activate the transparency channel in the Attribute Manager and switch to the “Transparency” menu. Set the color to blue. Set refraction to 1.4 and drag the material from the Material Manager onto the platonic object in the Object Manager.



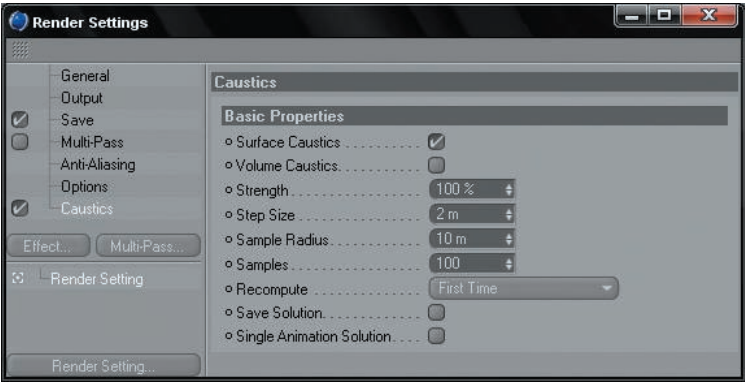
If we would have left the setting at 1 the light would have passed through the object unbroken without a caustics effect. Create a spot light (objects / scene / spot light). Place it at the following position: X=270, Y=300, Z=-100 at an angle of H=70 degrees, P=-25 degrees using the Attribute Manager. The light is now shining through the object onto the floor. Render the scene.



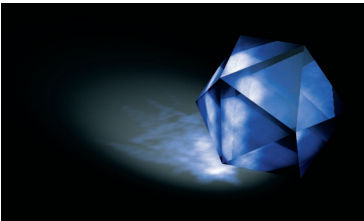
Part of the light is falling on the platonic, part of it is falling on the floor. Still there is no caustic effect anywhere in sight - the effect must first be activated. Select the light and switch to the caustics page of the Attribute Manager. Activate “surface caustics” for the light.



Open the Render Settings menu (Render / Render Settings) and select “Caustics” from the “Effects...” menu.



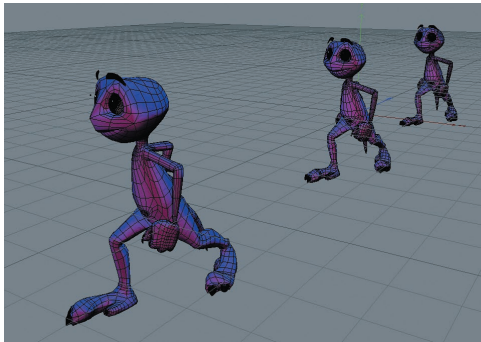
Render the scene. The caustics effect should now be visible on the floor.



6. Quick Tutorial – Depth of Field

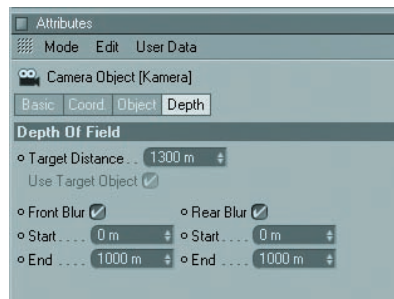
You probably are familiar with the depth-of-field effect from the field of photography. Depending on which lens is used in which combination of camera settings it can only be focused on one more-or-less small area at once – everything outside this area is out of focus. This effect is often used as a dramatic element in movies to draw the viewer's attention to a specific element in the scene. Of course CINEMA 4D's Advanced Render Module also offers this technique – as we will now demonstrate.

Load the scene "QS_DoF.c4d".



Create a camera object (objects / scene / camera). Place the camera at the following position: X=370, Y=200, Z=-1500 using the Attribute Manager,. Set the angle at H=25 degrees, P=-17 degrees. In the editor window's menu choose (objects / scene / camera). The editor view will switch to the camera's view. All three figures should be visible.

Now we'll make the settings for the depth of field. Switch to "Depth" in the camera menu of the Attribute Manager. Set the target distance to 1500 1300 and activate "Front Blur" and "Rear Blur" in the camera's Depth tab in the Attribute Manager.

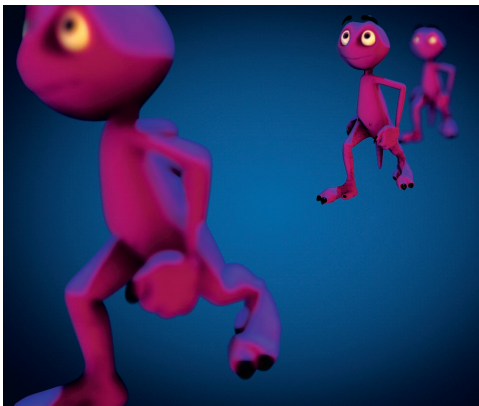


The target distance is the point of focus of the camera. An object located 1300 units from the camera object – as is the case with our figure located in the center – will appear the clearest. Objects located in front of or behind this object will be rendered out of focus. Simply activate this effect in the Render Settings to make this effect visible.

Open the Render Settings and click on the “Effects...” button. Select “Depth of Field” in the pull-down menu and increase the “Blur Strength” to 10%, activate “Background Blur” and set it to 20%. This makes the effect more obvious.



Render the scene. Only the figure in the center will be in focus. You can customize the strength of the effect using the parameters in the Render Settings.



7. Tips & Tricks

Rendering often requires you to make a choice between speed and quality. Especially scenes using Global Illumination or Caustics can take a long time to render. This is why we recommend that you experiment with the Parameters and to initially use low values. For example, set the Global Illumination “Diffuse Depth” value to “1” to begin with and make a test render. If the illumination provided by GI is still too dark, increase the “Diffuse Depth” value gradually until you achieve the desired result. However, avoid raising this value above “4” since this will increase your render times enormously!

Make generous use of compositing tags. This makes it possible to reduce the exactness of the rendering, thus reducing rendering times greatly.

As you already know, CINEMA 4D allows you to animate effect parameters so that you can make changes to them at any time. Imagine caustics that change as you wish or the focusing of a camera.

Load a bitmap image in place of a color into the luminance channel of a material that you use to illuminate a Global Illumination scene. The objects will then be lit with the colors of the image. This looks especially good if you use HDRIs. HDRI is an image format that contains special image luminance information and is thus an excellent choice for this effect. You can find information about HDRIs online.

You can pep up your scene even more with the post effects “Highlights” and “Glow”. Using “Specular” you can give your material’s highlights any form you wish, like a star, for example. “Glow” does just what the name says – it lets a material glow. Try it!



© Fran Pérez, www.digitalfran.com

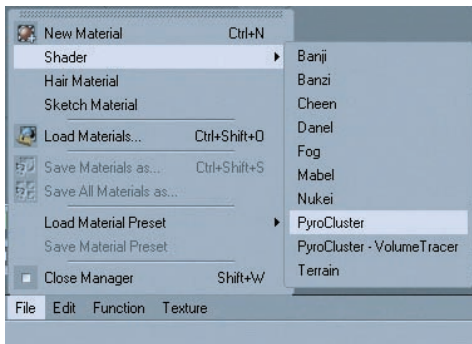
Welcome to PyroCluster

(Included in the optional Advanced Render module)

This is the Quickstart Tutorial for PyroCluster. It will lead you through the initial steps of this module and show you what's possible with this amazing 3D shader. Here is how this tutorial is structured:

1. Introduction

“PyroCluster” is a powerful tool for creating volumetric smoke, explosions and fire effects. It's a particle-based 3D shader that is generated by CINEMA 4D's integrated particles or the module's own particle system “Thinking Particles”. Of course the shader can be animated and can be found in the Material Manager under File / Shader after installation. More about that later.



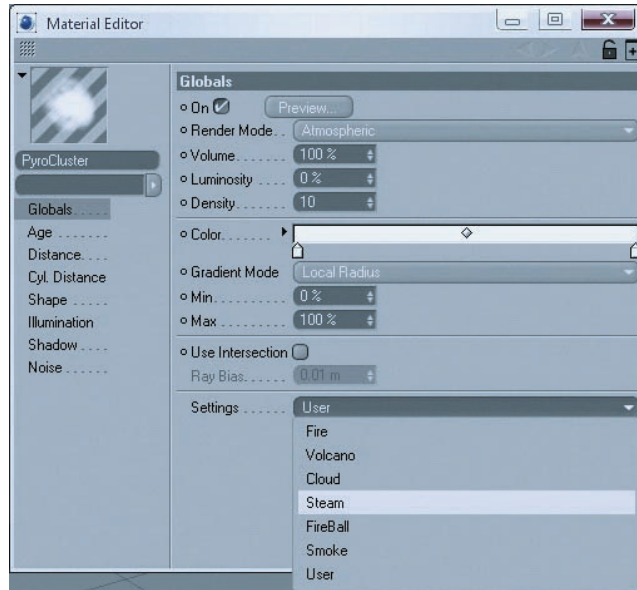
You can use this tool to create realistic effects just like in the movies. After you have worked through this tutorial you will realize that some effects are easier to create than you probably thought. As we mentioned, PyroCluster uses CINEMA 4D's particle system and is also animated using this system. Merely the look is determined by PyroCluster. This makes it possible for you to use all modifiers such as wind, gravity, turbulence, etc. to achieve the desired effect. Mixing and rendering several volumetric effects in one scene is a technology that lets you create even more effect combinations. If you're a lucky owner of “Thinking Particles” you have even more possibilities in the creation of your effects such as explosions. Whereas CINEMA 4D offers enough opportunity for you to forget time and space with its integrated particle system.

If you're one of the people who would rather hear the sound of the “director's clapboard” this would be a good time to go straight to chapter 4, “Quickstart Tutorial Part 1”. Here are a few basic facts about PyroCluster's interface for those hungry for more knowledge.

2. General Information / Interface

The actual effect basically consists of the PyroCluster shader, the PyroCluster volume tracer, an emitter and an environment object. Together they comprise the necessary elements who's properties you can set to create any and all pyrotechnic effects. The components in detail:

The **PyroCluster shader** is the material which is assigned to the emitter. We've taken a little work out of your hands by entering a couple of presets under "Settings" in the "Globals" menu of the PyroCluster's Material Manager.



PyroCluster's Material Manager menu is the quickest way of creating cool effects. Fire, Volcano and Cloud as well as Steam, FireBall and Smoke are at your disposal. This is where you set the gradients and can determine the self-illumination properties and shadow parameters.

The **PyroCluster volume tracer** produces volumetric effects and is assigned to the environment object. Some of the settings you can make here are Render Mode, World Step Size (One of the most important parameters. Read more about it in chapter 6 "Tips & Tricks"), Volume Light, Ray Trans. Limit, etc.

CINEMA 4D's own **emitter** is part of the particle system. It's an object that emits particles which you can then modify as you wish. The number of (Birthrate), velocity, lifetime and rotation of the particles can be set.

Last but not least is the **environment object**. You can add fog or an environment color to your scene. In our case it will serve to incorporate the PyroCluster volume tracer as a material so that it can assume the scene's volumetric calculation.

"Make Preview" is a helpful feature that gives you the chance to examine your effect before you render the entire scene. This saves a lot of time, of which we all have too little. We also gave PyroCluster various fractal types you can fine-tune according to your needs. A cumulus cloud, for example, would require a different type of fractal than would a gas cloud. You can use yet another fractal to depict coral or stone formations to name only two of many.

You can display the execution and “development” of an effect in the form spheres, cubes or other objects. This gives you an excellent overview of the effect and keeps you from having to render the entire effect just to see how far a trail of smoke has traveled. Of course all PyroCluster effects can cast shadows on other objects, if desired, using the normal scene lighting. In addition, self-illumination can be used for fire effects, for example, which will be included in the Global Illumination calculation (similar to “normal” self-illuminated materials – Advanced Render Module).

I think we’re done with the “dry run” and can get down to business: the Quick-Tutorial. Using a relatively small scene we will demonstrate how quickly and easily you can achieve your desired results. After working through this tutorial you will have the skills to create your very own effects.

If you are just about to pour yourself a cup of coffee or lean back and relax a little – take some time to look at the following sample images. They will give you a good impression of what PyroCluster can do.

3. Sample Image



© Tobias Deml, thegentle0@gmail.com

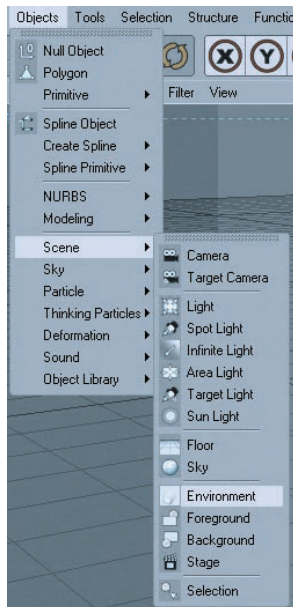
4. Quick Tutorial – 10 Steps To Glory

This tutorial describes the 10 basic settings needed for a PyroCluster effect. Once you have completed this exercise you can play with the settings of the example scene or set your own properties as you wish to see what settings do what to which effect, etc..

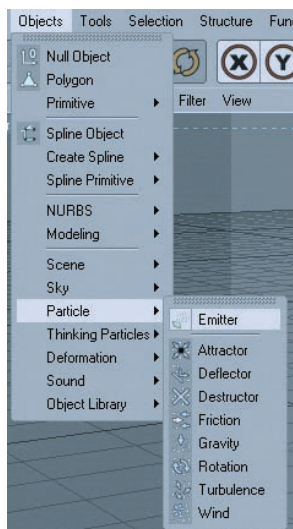
1. Open a new (empty) scene in CINEMA 4D's main menu (File / New).
2. Select "File / Shader / PyroCluster" and "File / Shader / PyroCluster-Volume" in the Material Manager below.



3. Create an environment object (Objects / Scene / Environment).

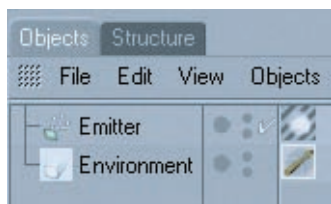


4. Create an emitter (Objects / Particle / Emitter).

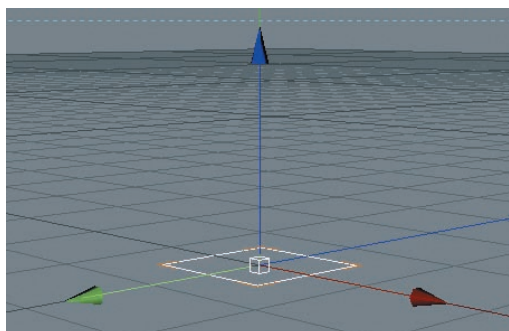


5. Assign the PyroCluster material to the emitter.

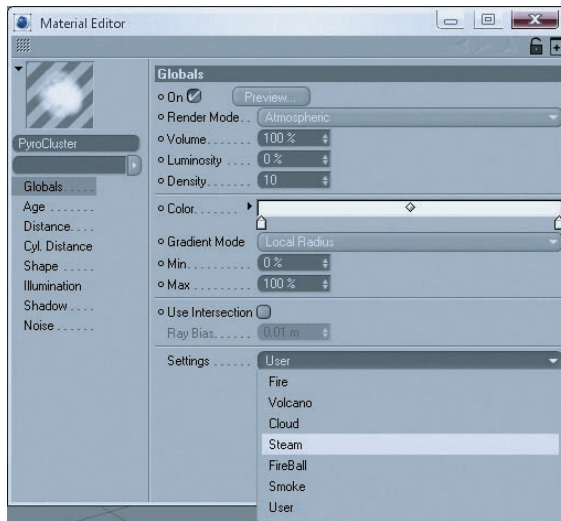
6. Assign the Pyrocluster Volume Tracer to the Environment object.



7. Set the emitter's "P" angle to 90 in the Object Manager's Coordinates Manager.



8. Double click the material “PyroCluster” in the Material Manager and select “Steam” in the “Global” menu.

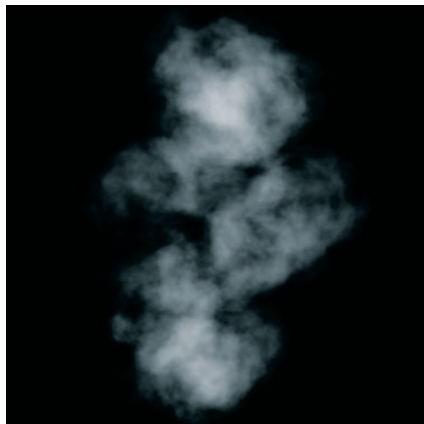


9. Answer “yes” to the question “Do you really want to convert parameters?” and close the Material Editor window.

10. Drag the time slider to frame 40 and select “Render / Render View”.



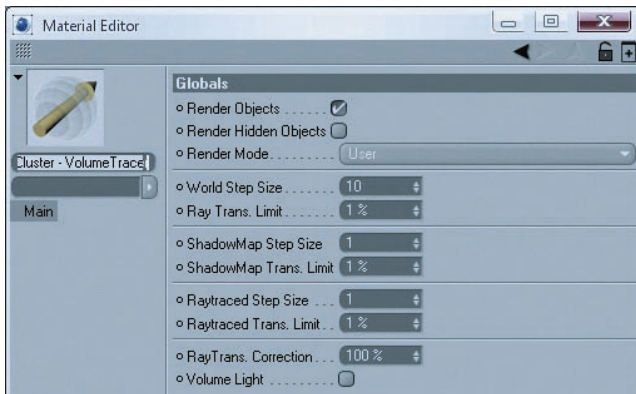
Now sit back and finish that cup of coffee.



Voila! If you go ahead and test the “Fire” preset you could qualify as a junior pyromaniac and no court in the world can lay a finger on you! This scene was calculated relatively quickly but you will eventually notice that it can take longer depending on the complexity of your scene and the settings you use. There are a few tricks you can use to reduce render time without losing quality. If you want to optimize the settings right now (before we move to the next step) we suggest you flip to the “Tips & Tricks” chapter and check back with us later. Go ahead, we’ll wait.

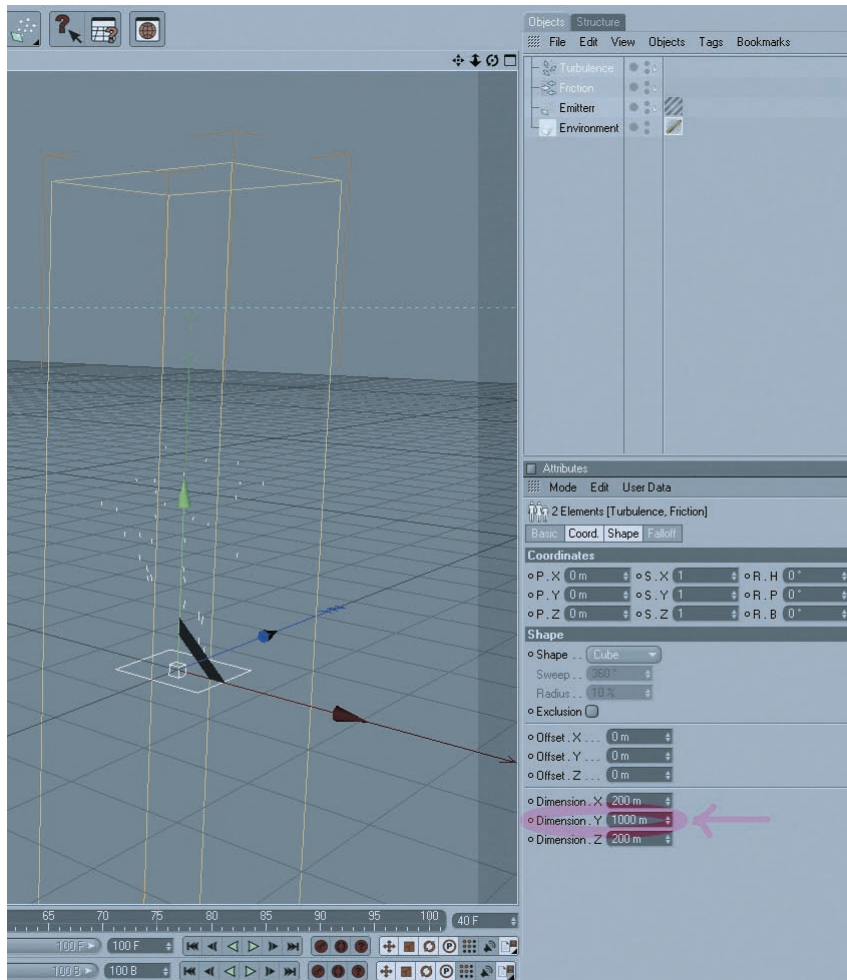
5. Quick Tutorial – Optimize and Animate

Let’s tackle the part of the tutorial that will show us how to make the steam effect more realistic and show us how it all looks when it’s in motion. Move the editor view until the emitter sinks below the bottom edge of the editor. Before we generate our first animation we will have to modify our PyroCluster-Volume tracer a little. Double click the volume tracer material in the Material Manager and set the render mode to “User” and raise the “World Step Size” to 10.

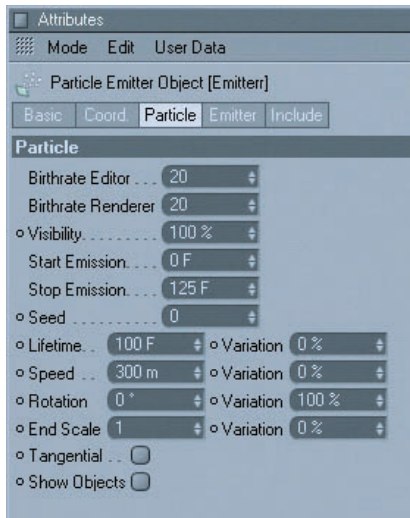


You will see why we changed the last setting in the next chapter. Select Render / Make Preview”, and set the frame rate to 15. Click on “OK”.

Now you have created your first PyroCluster animation with just a few clicks. You can see how quickly you can go from an empty screen to an (almost) finished animation. We can use two objects to pep our scene up a little as well. This is how we will do it: We will create a turbulence and a drag object using CINEMA 4D’s particle system (objects / particle / friction and turbulence) and give both a Y-size of 1000 in the Coordinates Manager (make sure that you increase the Y-Scale in the Attribute Manager to “5” for each object. Otherwise all object coordinates will be increased parallel to a value of 1000! Alternatively you can simply change the Scale value in each object’s “Shape” tab in the Attribute Manager). The editor view should now look like the image below



Raise both of the emitter's particle values (Birthrate Editor / Birthrate Renderer) to 20 (Click on emitter in the Object Manager and select the "Particle" menu). Reduce the Lifetime from 500 to 100 and the Speed parameter to 300. Switch to the "Emitter" and set Angle Horizontal and Angle Vertical to 30. The friction object gets a strength (friction-coefficient) of 4.



Now you can generate another preview (Main Menu / Render / Make Preview).

The steam's behavior is much more realistic now. The particles are being emitted faster but have a shorter lifetime and the wider angles disperse the particles by 30 degrees. The turbulence object adds (you guessed it!) turbulence and the friction object slows the particles and can even bring them to a complete stop if the value is raised sufficiently. Of course you can try your own or additional settings. Try different settings and see which ones work best for you and your scene. Our new settings have completed the scene and the viewer can recognize the effect as steam.

Before we show you the tricks we have up our sleeves that will speed up your renderings we recommend that you inform yourself about the "Gradients" (Color gradations) (double click "PyroCluster material; "Age" menu). Many of the parameters are controlled by these gradients. These let you not only influence the color but also the animation, look and properties such as Age and Size of a particular effect. We are dealing with a very powerful function here which influences the so-called Voxels during a particle's lifetime. The "Old Radius", for example, changes the Voxel's expansion between birth and death. You can also determine which colors the volumes will have during their lifetime. There is an unbelievable number of possible settings for you to try.

Have you had a chance to take a look at the "Tips & Tricks" chapter? If so, you have probably been able to save a lot of render time in the course of this tutorial. At least everyone else will finally soon find out about the most important part of this tutorial: How can I make my day nicer? Or:

6. Tips & Tricks

There are several methods to optimize your rendering but the most effective method is to increase the PyroCluster Volume's "World Step Size" parameter! Here you will also find several settings for the render mode. If you click on the drop-down menu you will see "User" on the list. Selecting "User" activates the "World Step Size" editing parameters. Double click the Pyrocluster Volume Tracer. Set the render mode to "User". Increase the "World Step Size" value to 20. Close the Material Manager and render the scene.

The world step size sets the subtlety of the subdivision in 3D space and searches for PyroCluster Volumes in the scene. A lower value increases and higher value decreases the time it takes to calculate an image. So what you have to do is find a value you can live with. Raise the value and watch the result. You can most likely improve the render time dramatically without compromising quality. It sounds unbelievable but by raising the value you can decrease the render time by a factor of 100 (or more). In chapter 4 of the PyroCluster manual “Working with PyroCluster” you will find a very good solution to finding a value you can live with. Play with the value and decide for yourself what’s acceptable for you. You can also raise the “Ray Trans. Depth” value to speed up your rendering.

Reducing the number of particles is a further method of optimizing render time. We suggest you use fewer particles and increase the size of the Voxel instead. You will notice that you will get the same result in most scenes with the exception of scenes which contain fast animation movements or a wide array of particle systems.

Less light is more! Of course we’re not talking about the actual look of the scene. You can never have enough light in a scene as long as it’s done correctly. If you can do without multiple lighting of a PyroCluster effect in your scene, then you should deselect the “PyroCluster Illumination” and “PyroCluster Shadow Casting” check boxes for some or all lights. The more lights included in a PyroCluster effect, the longer the scene will take to render. Sometimes it can even become unbearably long.

7. In Closing

You saw how fast you can generate impressive effects using PyroCluster and we think it’s time you tested its limits on your own, just to see what each effect does. You have practically no limits (and no court in the world can... oops, we already went through that!) If anything about PyroCluster is unclear just flip through the PyroCluster manual and try everything out.

Just remember that this demonstration is just one of many ways of generating this effect and we adapted it to the “restrictions” of this tutorial to make the process understandable easy to understand. Of course you would go much more into detail if you were generating this effect for a big-screen movie, including doing research on the behavioral characteristics of steam. Let’s take that coffee you were drinking earlier as an example. Is it still steaming? Take a good look and try to notice the details of how the steam rises, then simulate it with PyroCluster. Reality is the best example and a good power of observation is one of the most important factors in achieving the desired “Hollywood” effect. Isn’t it scary to know that you can generate anything you imagine using this module?



Welcome to Sketch and Toon (Optional module)

This is CINEMA 4D's Quickstart Tutorial for its Sketch and Toon module. It will introduce you to some of this renderer's fantastic creative possibilities.

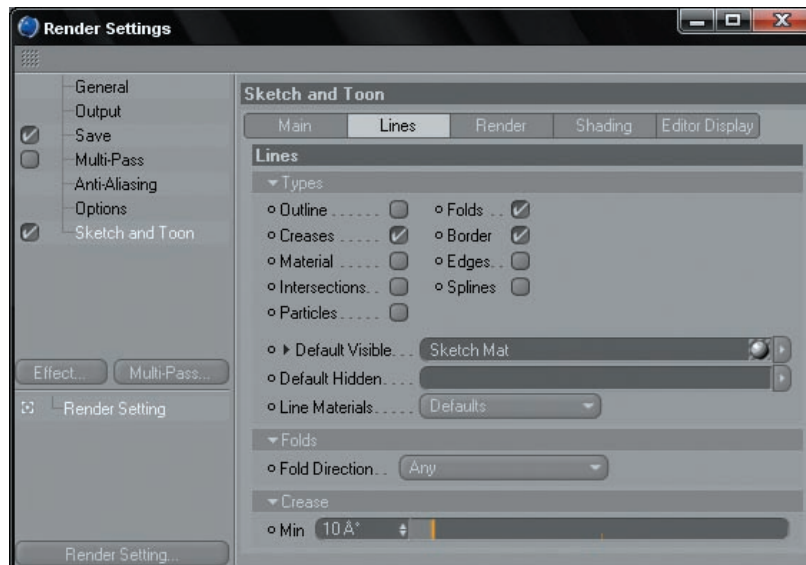
1. Introduction

Sketch and Toon belongs to the NPR family. This is an acronym for "Non-Photorealistic Renderer" and simply means that it's not the renderer's intention to generate highly realistic images but to do exactly the opposite: To generate images that look like they were created using "traditional" animation techniques. Do you want your scene to look like a technical blueprint or maybe a pencil sketch? Do you want to give your animated characters that traditional animation look? No problem for Sketch and Toon!

Sketch and Toon is very easy to operate. For instant results you can fall back on one of the many presets Sketch and Toon provides. Otherwise you can have a ball with Sketch and Toon's different settings. You will quickly realize that Sketch and Toon is a very powerful tool that offers limitless possibilities. You can take influence on practically any imaginable parameter. This gives you a cornucopia of render styles and you will never break the boundaries of Sketch and Toon – we promise! If you want to get a taste of how you can metamorphose your images just skip ahead to our gallery.

2. General Information / Interface

Sketch and Toon is a render effect. As you would expect you can find its settings in the Render Settings (Render / Render Settings) "Effects" menu. As soon as you have selected "Sketch and Toon" from the "Enable Post Effects" menu a wide array of settings is put at your disposal. These parameters let you determine the basic look of your Sketch and Toon renderings.



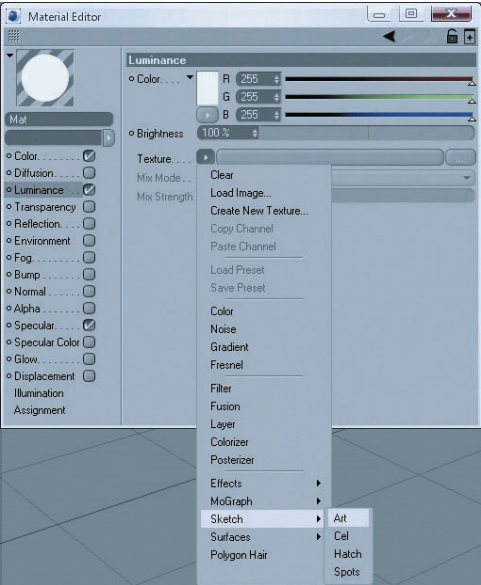
There's more. You will also find Sketch and Toon elsewhere within CINEMA 4D. Take a look at the Material Manager. A sketch material is created as soon as the Sketch and Toon effect has been activated. This is a material especially for the depiction of contours in a Sketch and Toon rendering.



You will also see that there are many more settings in association with this material in the Material Manager. These settings are global and affect the entire scene.

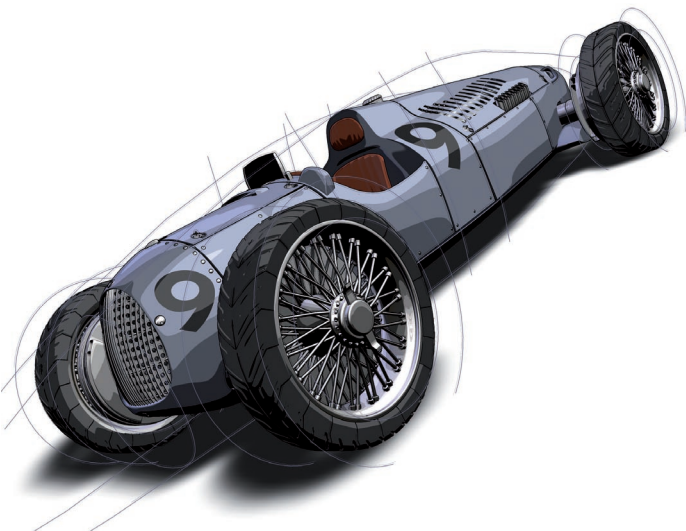


Keep this general rule in mind: The sketch attributes in the Render Settings determine WHAT will be rendered (contours? Hidden lines?), and the material attributes determine HOW something will be rendered (Line width? Line color?). Of course not every object has to be rendered in the same style. You can create an arbitrary amount of sketch materials and assign them to different objects. This makes combining a pen drawing with a cartoon object no problem. After all, Sketch and Toon offers four "Sketch Shaders". These can be placed into the channel of a "normal" material just like any other shader. We suggest you place them into the luminance channel. These shaders work independently, which means Sketch and Toon does not have to be active. The shaders are; "Art" for effects such as oil or acrylic painting, "Cel" for a cartoon-like style, "Spots" for a halftone print effect and "Hatch" for cross hatching.

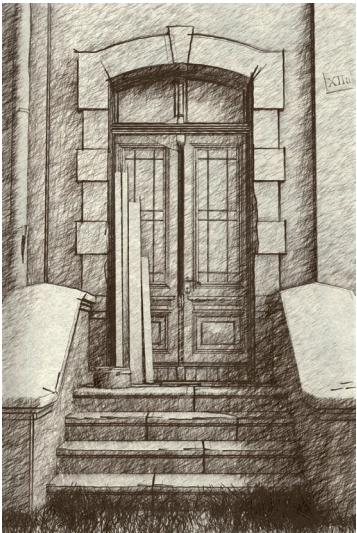


3. Sample Images

Here is a sample of the beautiful imagery you can create with Sketch and Toon.



© Marco Weiss, www.black-graphics.de



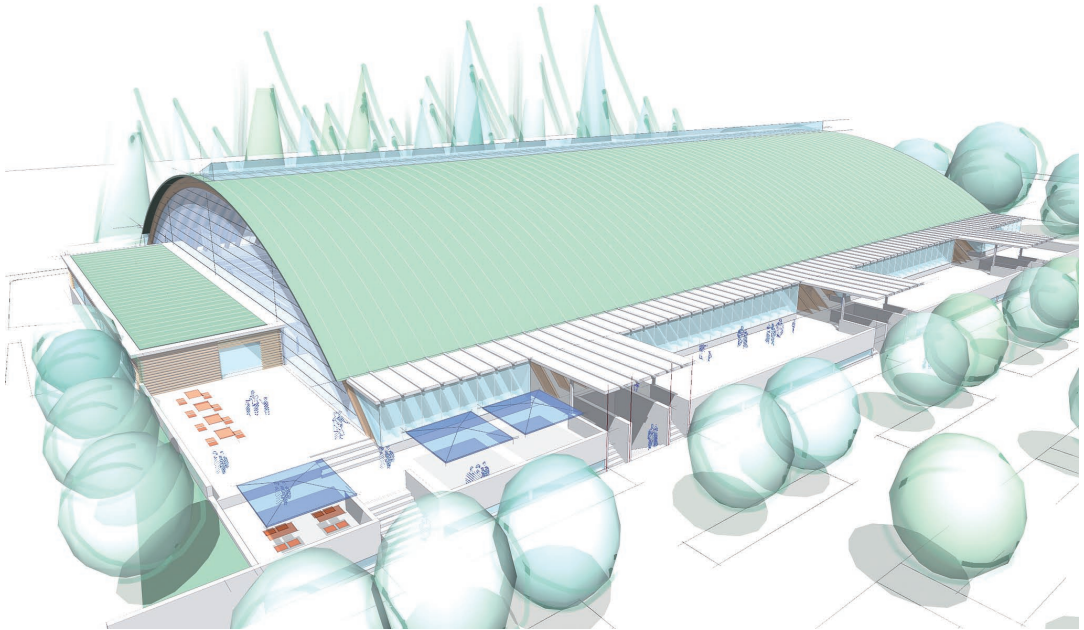
© Pavel Zoch



© P. Hofmann, M. Hilbert (pexel@3dup.com)



www.bonewire.de



© www.areyouplanning.de

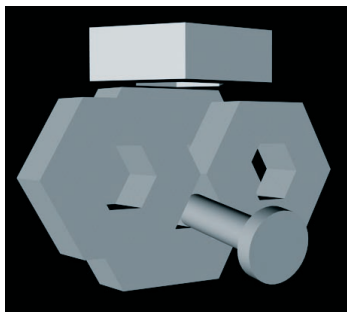


© Michail Stehle Design

4. Quick Tutorial – Outlines

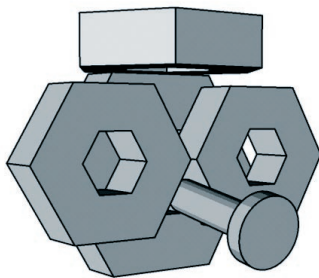
The moment of truth has come. Once you have worked your way through this tutorial you will have but a first impression of what you can do with Sketch and Toon. Sketch and Toon is a module that practically seduces you to experiment with it – give in to its seduction!

Open the file “QS_Outlines_Start.c4d”. This scene is comprised of a head made up of just a few primitives.



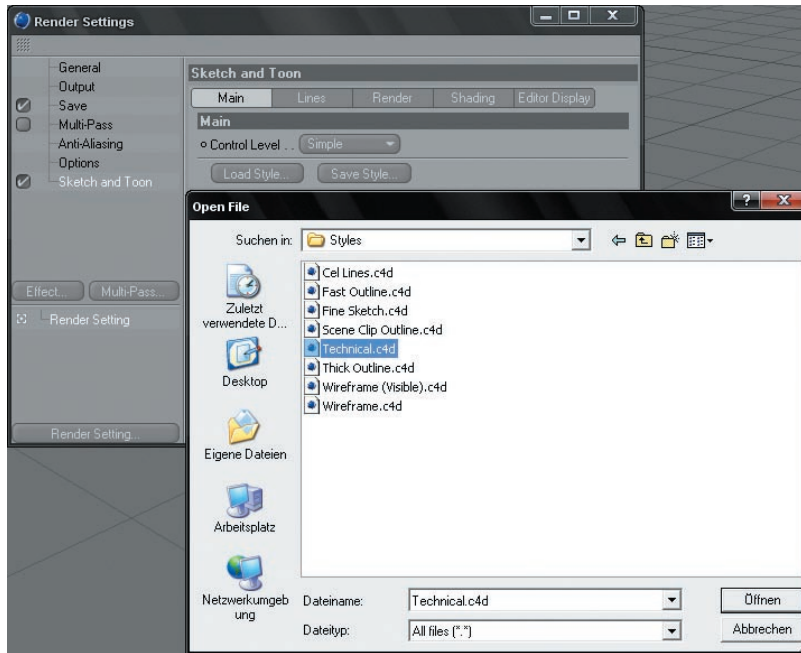
Let's have a look at how this all looks once it has been rendered with Sketch and Toon.

Open the Render Settings (Render / Render Settings), click on the “Effects...” and add a “Sketch and Toon” Post Effects menu. Render the scene.

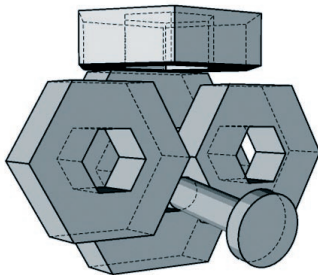


The result looks much different from the first rendering. All objects have a black contour and the color gradations are somewhat “stepped”. Notice that a new “Sketch Material” has been generated in the Material Manager. The look of the contour is defined in this material.

Switch to the “Effects” menu within the “Render Settings” menu and click on “Load Style”.



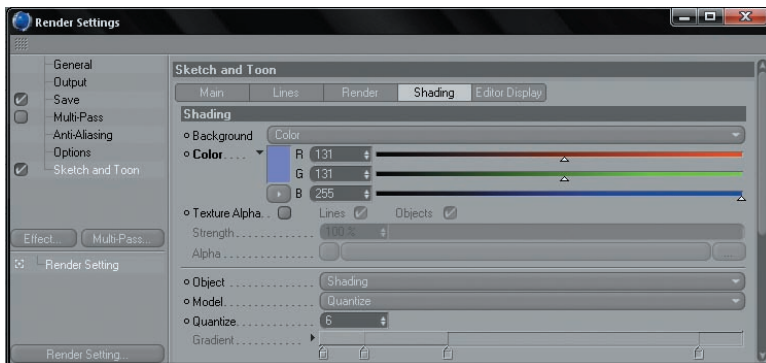
Navigate to your CINEMA 4D directory. You will find a number of presets under Library / Sketch / Styles. Select “QS_Outlines_Final.c4d”. This style is good for technical illustrations. Render the scene.



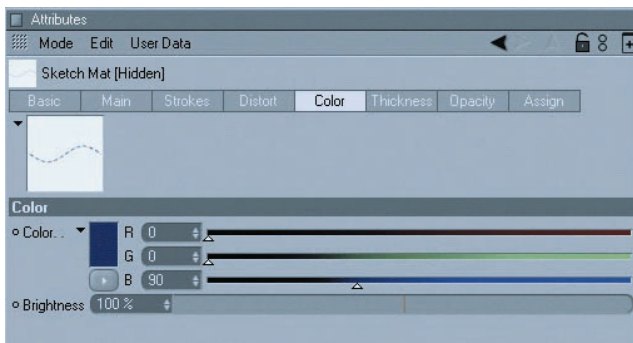
Again the rendering looks different. Now the lines which are normally hidden are displayed as dashed lines. Along with the preset two additional materials were placed into the Material Manager – “Visible” and “Hidden”. “Visible” defines the look of the visible lines and “Hidden” that of the hidden lines. The “Sketch Material” is no longer being used.



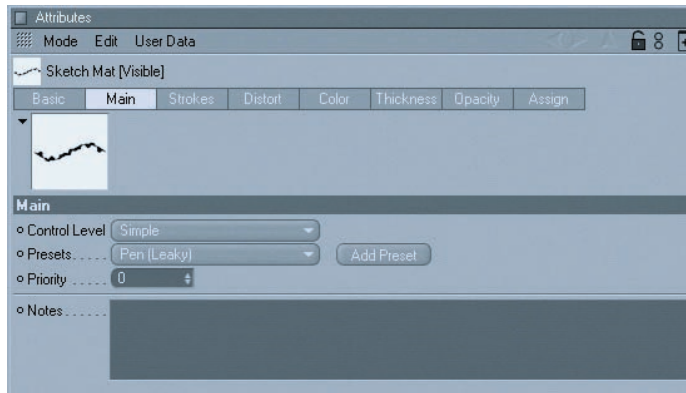
The Sketch material's preview display can be changed by double-clicking the material and selecting a line preview (right-click on the preview image). In the following menu you can select how you want your preview to appear (e.g. as a sphere). Let's bring some color into play. Switch to "Shading" in the Sketch settings (Render Settings / Sketch and Toon). Use the color sliders to set the "Background" color to a discreet blue (or your favorite color).



Select the "Hidden" material in the Material Manager and switch to "Color" in the Attribute Manager. Use the color sliders to set the color to a dark blue tone. Render the scene.



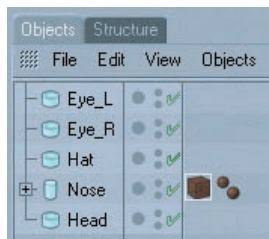
We can use the Render Settings to influence the entire scene and the material settings to specifically affect the hidden contour. Select the "Visible" material and switch to "Main" in the Attribute Manager. Select the "Pen (Leaky)" style in the "Presets" menu. Render the scene.



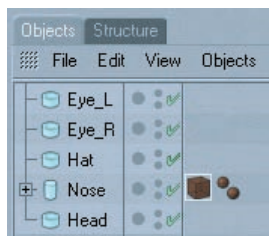
The visible lines already look like they were drawn with a leaking fountain pen and the hidden lines are still dashed. Go ahead and try out more of the presets!

Until now changes to the sketch settings have had an effect on all objects in the scene. We can also choose to change the look of only one specific object.

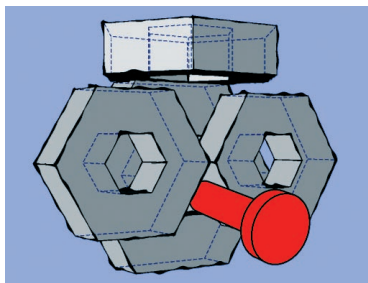
Select File / Sketch Material in the Material Manager. Drag the material from the Object Manager onto the "Nose" object. A "Sketch Style Tag" will appear next to the object. Select this tag.



You will see several settings in the Attribute Manager that you may already know from the Render Settings. The difference here is that the tag's parameters only effect on the "Nose" object. Switch to "Shading" in the Attribute Manager. Activate "Enable" and select "Custom Color" in the "Object" menu below. Set the color to red and render the scene.



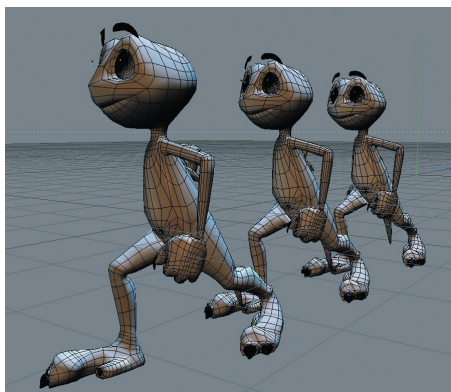
The nose is now red with a black contour.



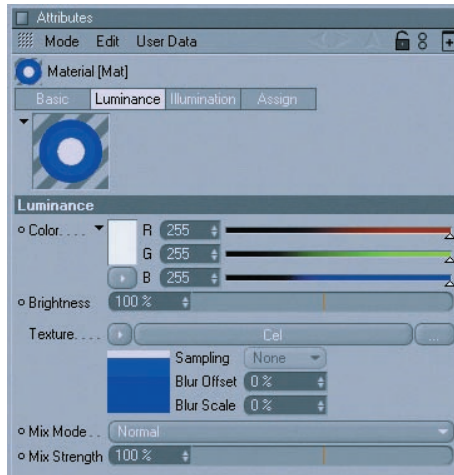
5. Quick Tutorial – Shaders and Tags

In this tutorial we will combine an object rendered in the sketch mode with an object rendered in the “normal” mode in the same scene. We will also examine the “Cel” shader, one of four special sketch shaders. Our goal will be to assign different render styles to each of three objects. The first object will only have a contour and it should look like it was hand-drawn. We will test the cel shader on the second object and give it cartoon-like colors. The third object will look like a normal CINEMA 4D object rendered with standard settings. All of these objects will be depicted in the scene together.

Open the file, “QS_SaT_Start.c4d”.

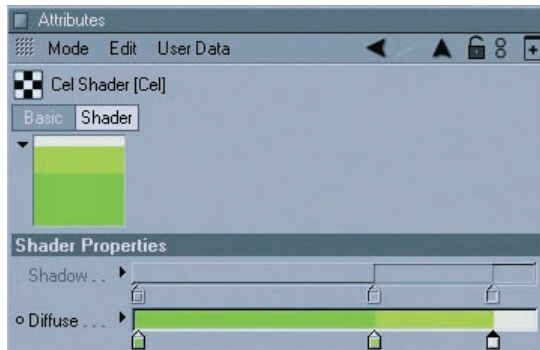


Create a new material in the Material Manager (File / New Material). Deactivate the “Color” and “Specular” channels in the “Basic” menu of the Attribute Manager. Activate “Luminance” and switch to the “Luminance” menu. Click on the small arrow to the right of the word “Texture”. Select Sketch / Cel.



The cel shader is a type of gradient shader. It can be loaded into any channel but for a cartoon-like look it should be put into the “Luminance” channel where its colors will always be rendered as full-tone and will not be influenced by the scene’s lighting.

Click on “Cel”. The cel settings will appear. 3 shades of blue are preset here by default but they don’t really match our figures. Double click on the color sliders and create three new gray-tone colors.

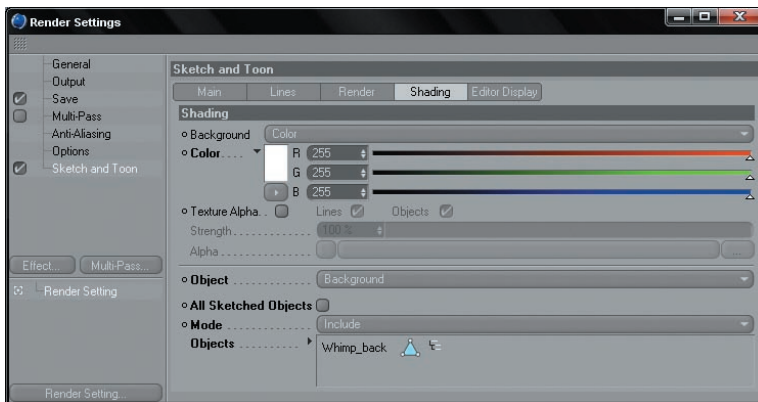


The number of colors represents the number of color steps an object will be colored with. 1-2 colors is always good for a cartoon look. Drag the new material in the Object Manager onto “Whimp_center” and render the scene.

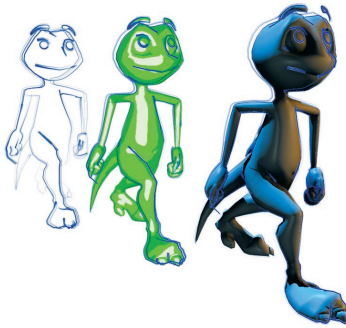


Our Sketch and Toon figure is standing between two “normal” rendered figures. It is not necessary, though, to activate the Sketch and Toon render effects in order to use a sketch shader. Now we will beautify the rear figure with a contour.

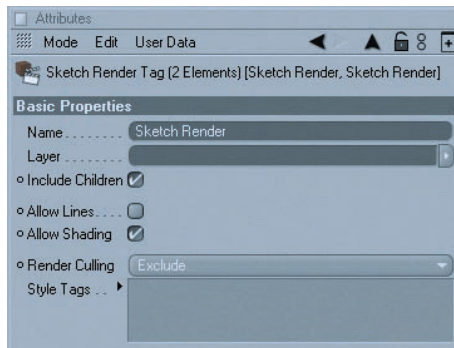
Activate the Sketch and Toon effect in the Render Settings. Switch to the “Shading” menu and select “Background” in the menu next to the word “Object”. Only the contour should be visible on the rear figure so we will set its “Inner Color” to the same as the background color. If we were to render the scene now the other figures would appear white as well so we will tell sketch that these settings should only apply to the rear puppet. Deactivate the “All Sketched Objects” parameter. Select “Include” next to “Mode” and drag “Whimp_rear” from the Object Manager into the text field under the menu. The shading settings will now only apply to this object.



Select the Sketch effect that was just added and select “Pencil (Soft Sketch)” in the Attribute Manager’s Preset menu. Render the scene.



Even though all figures have the correct “filling” each of the two front figures has assumed the contour style of the figure behind it. We will deactivate the contour rendering for these two objects completely. Select “Whimp_front” and “Whimp_middle” in the Object Manager and select Tags / Sketch Tags / Sketch Render. Deactivate “Allow Lines” in the Attribute Manager.



We’ve done it! Three objects, each with completely different render styles coexisting peacefully in the same scene.



6. Tips & Tricks

Sketch and Toon offers several levels of control. If you are taking your first steps with this module and are a little overwhelmed by the countless buttons and sliders simply set the level of control to “Easy” (either in the Render Settings or in the sketch material attributes). Several functions will then be “hidden”. Once you feel more comfortable with Sketch and Toon (which shouldn’t take too long after having completed this tutorial) you can move to the next level.

The time it takes to render a Sketch and Toon scene depends on the size and / or complexity of your scene. If it takes too long, check for objects in the scene that can be excluded from the rendering process using tags. Lowering the anti-aliasing quality (contour smoothing) can speed your rendering up as well.

Make use of Sketch and Toon’s flexibility. You can assign different styles to visible and invisible lines, just as you can assign a different style to individual objects in your scene. The combination of standard rendering and Sketch and Toon can result in especially spectacular images.

Use the countless presets, especially in the beginning, to give yourself a good working knowledge of Sketch and Toon. If you find a preset that you really like, examine it closely and see what all it can do when you apply your own settings. This is the best way to find out how Sketch and Toon “ticks”.

Welcome To NET Render (Optional module)

Have you ever asked yourself “What should I do if my CINEMA 4D animation will render far longer than my life expectancy, regardless of any and all medical advances?” The answer is “NET Render!” This Quickstart Tutorial will help you reduce the time you have to wait for your renderings. First let’s look how this tutorial is structured:



1. Introduction

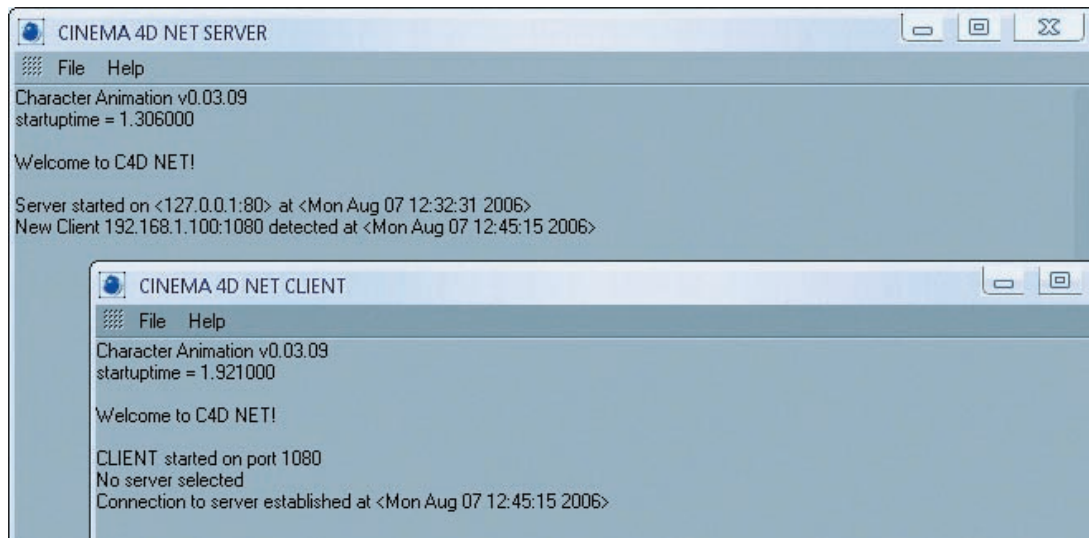
Your computer probably won't be able to handle most large and complex animations by itself in the time you need. NET Render lets you use every additional computer you have which is connected per TCP / IP in the render process. Better yet is the use of an entire render farm or of available company computers already connected to the network. CINEMA 4D's Studio Bundle offers you the possibility to connect an almost endless number of computers to the network to render your project using NET Render.

2. General Information / Interface

The software package consists of two components: the **server** and the **client**. Simply put, you send the server a job and it distributes them to the clients who do the work and send the completed job back to the server. The server distributes the jobs efficiently and intelligently, meaning that the entire network will not be slowed by the slowest computer. In addition you can add or remove individual clients during the rendering process. The new capacity is registered with the server and includes these in the render process immediately. Even the client program reacts intelligently and is, of course, multi-processor capable.

3. Sample Images

Here are a few screenshots of the NET Render interface:



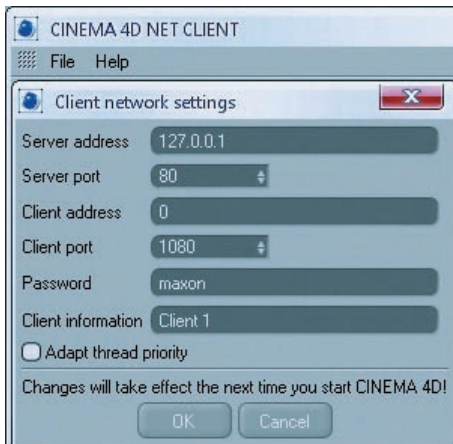


4. Quick Tutorial – Installation / Interface

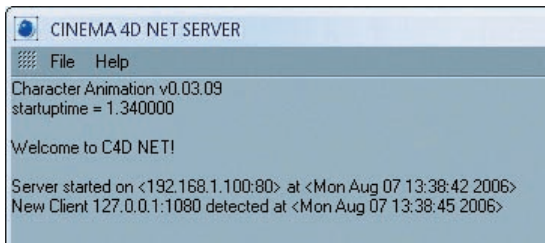
Let’s assume you have 4 computers at your disposal which are all connected to a TCP / IP network. Computer Number 1 (subsequently we will rename our computers “Computer 1”, “Computer 2”, etc.) is your main computer from which you will be distributing the work. The remaining computers will do the rendering. To let you continue to work on your CINEMA 4D scene without losing computing capacity on your computer, Computer 1 will not render at all. Of course this is only one of many possible configurations. Proceed with the following steps.

Install the NET Server on Computer 1. (When you run the installation file you will eventually be asked which of the two NET Render components you would like to install. In this case the installation of the NET Server will be enough! Install the NET Client on Computer 2 through 4. Now you have to find out the IP address of the computer on which the NET Server was installed. (In Windows XP: Start / Run; enter “cmd” and confirm with the “Return” button on your keyboard) The top IP address should be the correct one. Mac OS X users will find the IP address under “System Settings / Network”).

Start all three clients, Computer 2 through 4. Enter the IP address we just got as the “Server Address” in the “Network Settings” for all three computers. After you have done this restart all clients.



Open the NET Server and see what happens. If you did everything correctly all three clients should appear in the NET Server’s main window. Each client has now communicated with the server and, after a brief “hello” has been registered as “New Client”. You have just completed the most difficult part. If server and client can’t find each other check the NET Renderer’s “Problem Solving” chapter.



Open your web browser and enter the previously ascertained address (in our example, this would be <http://192.168.1.100:80>). Confirm your entry by pressing the “Return” key. If the CINEMA 4D NET Render homepage appears in your browser the connection was successful. If not, read the troubleshooting section of the integrated CINEMA 4D Help system.

This is the default IP address you must use when the NET Server has been installed on your computer, from which you can also distribute jobs using the browser. If the working computer has been installed onto a “render slave”, though, you would simply enter the IP address of that respective computer. It’s not imperative that you install the NET Server on your working computer. Since the server barely uses any resources, any computer connected to the network can be implemented.

5. Quick Tutorial – Jobs and Administration

This is part 2 of the tutorial in which you will learn job and user administration. Click on “Enter” in your browser window. If you are prompted for a user name and password use “administrator” as the user name and *leave field empty* as the password. This is the default access code. If you still can’t get in, open “Server.ini” in the “serverprefs” folder in CINEMA 4D’s main directory where the user data is located. (Windows: Open with Notepad; Mac: Open with TextEdit).

Under Windows XP you will find the “serverprefs” folder in your user data directory under “c:\Documents and Settings\[user name]\Applicaton Data\MAXON\NET Render R11_[version]\serverprefs”. **Under Windows Vista:** “c:\Documents and Settings\[user name]\App Data\Roaming\NET Render R11_[version]\serverprefs”. **Macintosh:** “Macintosh HD/User/[user name]/Library/Preferences/MAXON/NET Render R11_[version]/serverprefs”.

Note that the aforementioned Windows folders are hidden by default. They must first be unhidden manually (Folder Options / View / Show hidden files and folders).

Make sure this file is only modified if absolutely necessary! No other persons should be given access to these files. Otherwise they will have access to all user names and passwords and will be able to access render jobs!

You should now be on the “jobs” page. We will get to the “client” and “users” pages later. Enter a unique project name in the text field and press “Create New Job” to confirm. This new job should be listed under “Inactive Jobs”. Click on the job you just created to get to the next page. On the left you will see an “Upload” button. By clicking this button you can upload your CINEMA 4D project files. Click on “Search” and select your CINEMA 4D file (and of course the corresponding textures). You can select up to 10 files. If your project contains more than 10 files simply repeat the process or read Tips & Tricks. Now click on the “Upload” button. We are now again on the “Details” page where the project files we just uploaded are now listed. All you have to do now is click on “Start” next to your job name on the “Jobs” page and you’re done.

The progress of your rendering is shown in % on the same page in your job’s column. This offers you a constant overview of which job is finished and which is still being rendered.

You can go to the “Clients” page and see the clients you have started and information about which clients are busy and which are being lazy and are doing nothing. If the rendering has been completed simply click on your job’s name to see the rendered images (on the right under “Result Files”). Select the desired file and click on “Download”. When you have finished this tutorial, please read the chapter “Tips & Tricks” where you will find some important information such as “Where should I save my textures so every client can access them?”.

That’s it! Now you see how easy it is to operate this module, even if the name NET Render seemed a little forboding at first. We want to make life as easy as possible for you with this module as well so you can concentrate on more important things than complicated program settings.

Now we will touch briefly on the module’s user administration. Only the administrator can add or delete new users. Go to the “Users” page in your browser and enter a non-valid user name. (Remember that a job name can consist of maximum 32 characters – letters “A-Z”, numbers “0-9” and “_” are allowed.) Click on “Create New User” and set a password in the next window. (For security reasons, please inform the user of this password personally or encoded) The user can also be given administrative privileges. Of course a user can be deleted at any time or you can edit his or her user information. A user name can only be changed internally if a new user is created and the old one is deleted.

Now you understand NET Render’s basic information and know how it works.

6. Tips & Tricks

NET Render's major advantage is that it lets small companies and small budgets do network renderings! Even someone with only two computers can save time and nerves.

You must select only static TCP / IP addresses! It is especially important that computers with a dial-up modem internet connection use the "dynamic" setting. This setting must be set back when you want to reenter the internet. A better solution would be to install the NET Server on a network computer that already has a fixed TCP / IP address.

In order to avoid any unpleasant surprises, make sure every NET Client has access to all necessary files. Save the CINEMA 4D scenes that are to be rendered using the option "File / Save Project". Make sure all settings under "Render / Render Settings" are correct. It's not necessary, though, to save your work to a specific path, which brings us to our next issue: "Where do I find my finished work?"

You will find your rendered images or movie in the path you defined under "User / [your directory]" or "User / *installed user*" in the Job directory's "results" folder. Interim results (*.b3d files) can also be opened in the CINEMA 4D Picture Viewer.

If you have administrator rights you can download the finished files (from the computer on which the NET Client has been installed) from the regular file server onto your computer.

Since Release 9.5 it is possible to open intermediate results (*.b3d) in the Picture Viewer.

You say you'll be on vacation in the Bahamas and want to render your animation? No problem. Just turn on the external access to the render server which will allow you to control everything over the internet.

Flip through the NET Render manual for even more tips. Here you will find the most common problems and their solutions.



Welcome to Dynamics (Optional module)

This is the Quickstart Tutorial for Dynamics which will help you get to know the world of gravitation and physics. With the use of sample images and a two part tutorial we will give you an impression of what this powerful tool can do. Let's begin with how this tutorial is put together:

1. Introduction

You can become “Lord of the Masses”! We are referring less to your next diet than to the fact that you will be in command of so many virtual masses. Dynamics is the perfect extension module for imitating the behavior of physical bodies. Dynamics has made child's play out of a task that has been very time-intensive in the past. Simulations such as collision, drag and gravity are all done by Dynamics. This includes the virtual bowling alley and even such soft body simulations such as rubber are no problem for Dynamics. Of course the mixing and colliding of rigid and soft bodies is also possible. As you have become accustomed to with CINEMA 4D, this module is also fast and reliable (rigid and soft body simulations in real time). If this isn't enough for you, you can adjust any Dynamics simulation, depending on the project! Simply “Bake” the simulation and transform it into a real keyframe animation. Then you will have the freedom to correct every frame manually in order to make it fit your needs. Sounds promising, doesn't it? To keep it from just sounding good we will move right on to the next chapter to show you the basic interface so you can convince yourself.

2. General Information / Interface

In most cases, a pre-requisite for a Dynamics simulation is a solver object and two objects, each with a Dynamic Tag (Rigid Body Dynamic Tag, Rigid Body Spring Tag or Soft Body Spring Tag, depending on your needs) and one Effector, which is responsible for the most important physic, the gravitation. You will find all Dynamics objects in the main menu under “Dynamics”. You can also integrate the Dynamics tools for quicker access.

Select Dynamics in the main menu. Subsequently, click on the small dashed line at the top edge of the menu. This will undock the menu. You can now right-click on this menu and can customize it as needed. You can activate or deactivate the following functions by right-clicking on them: Text, Vertical (arrangement) and Icons/Icon Size. Make the menu easier to work with by deactivating Text, increasing the icon size and activating Vertical. Since a free-floating menu can be bothersome at times we will now integrate it into the layout. Simply click on the dashed line and drag the menu to the main Tool Palette. Voila, the Dynamics menu has just been beautifully integrated into your layout!

A given menu can be integrated into your layout wherever a black line appears when the menu is dragged over a particular palette or region. Once the black line appears just let go of the menu and it will be integrated at that location. You can use this method to customize your entire layout, which can subsequently be saved in the main menu under Window > Layout > Save Layout As... You can find the Dynamics Tag by right-clicking on the respective polygon object needed for the Dynamics function. Of course there are infinite combinations for such functions. You will get to know at least two of these in later chapters.

In order to make a simulation possible in the first make all objects, including gravitation, sub-objects of the solver object. Objects lying outside of the solver object will not be included in the simulation. This allows the creation of several optimized simulations within one scene which are completely independent of one another. If you make just a few changes to the properties of the Dynamic Tags of the polygon object, you'll have your very first Dynamics simulation – but we don't want to take the wind out of our Quickstart sails just yet. So we'll give you a few facts about each component and how they work together.

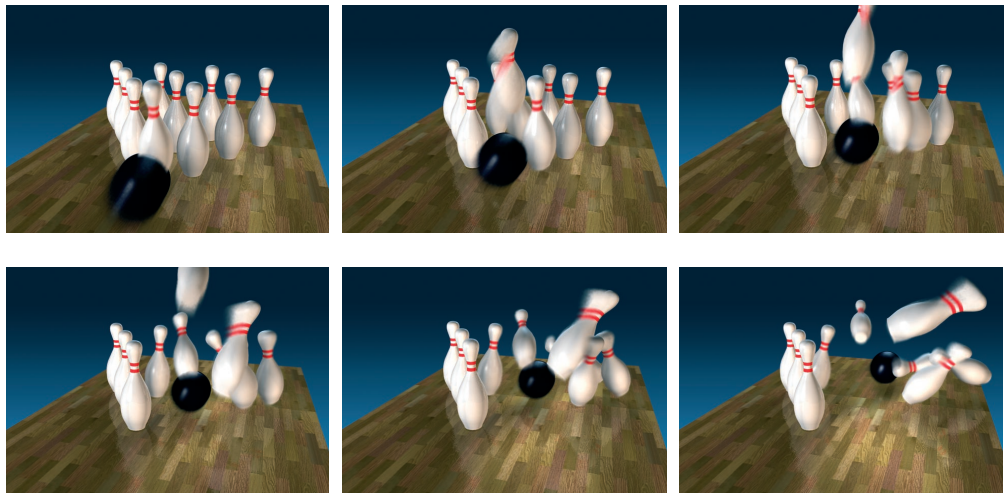
The **solver object** is the star of the Dynamics module. It takes over all calculations in dynamic simulations of its sub-objects and bodies. You determine the algorithm to be used and the distance between the objects that are to collide ($\text{Epsilon} > \text{Eps}$).

The **base object** refers to the object required for the Dynamics simulation. You can use parametric objects which have been converted to polygon objects or imported polygon objects.

The **Dynamics tags** give the object its character. They let a polygon object behave like a hard metal or wood body on the one hand or a rubber ring or pudding on the other. Only your fantasy is the limit! There are three types of Dynamics tags: The Rigid Body Dynamic Tag is for bodies with a hard surface. The Rigid Body Spring Tag lets you attach springs to polygon objects. Each point of an object can serve as a "spring". An example of this would be a figure hung on metal springs or the classic jack-in-the-box. The Soft Body Spring Tag is used mainly in soft bodies, for example rubber balls, or dog's ears. This can also be done using the CINEMA 4D module, Clothilde.

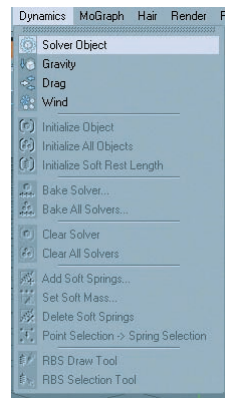
With effectors we have gravitation, drag and wind. You generate the gravitation and apply drag to the object, which forces the animation to end eventually. Without drag a rolling ball, for example, would never stop rolling and a rubber ball would never stop bouncing! If you can't wait to follow in Isaac Newton's footsteps, flip to the next chapter.

3. Sample Images

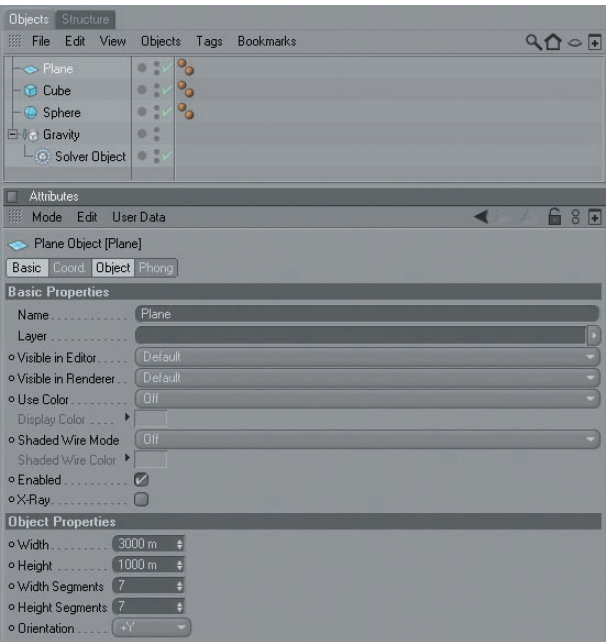


4. Quick Tutorial – Rigid Bodies

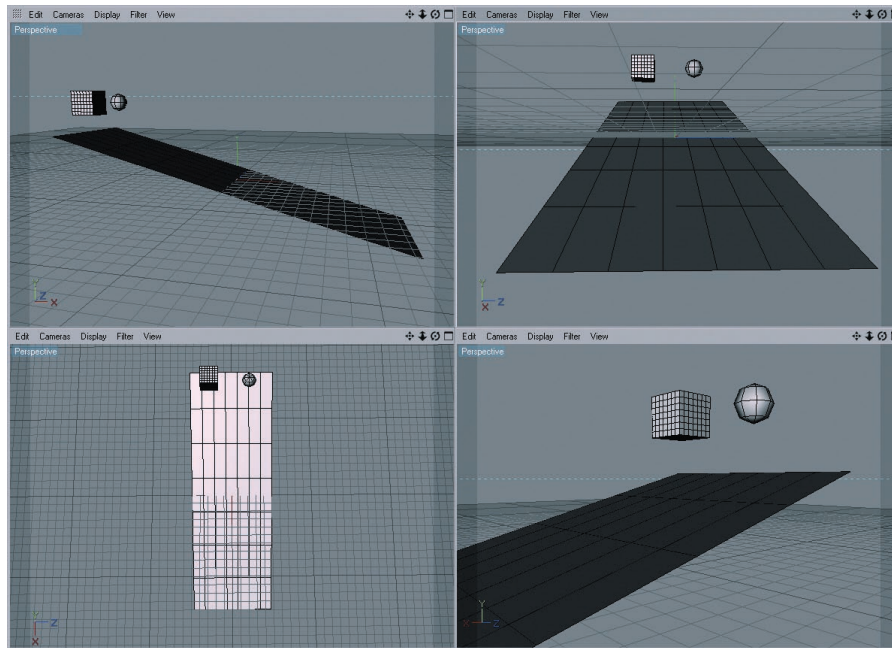
With just a few clicks of the mouse we will now create a scene in which several primitives will collide with each other. Let's begin by setting up the necessary objects. In the main menu, click on Plugins / Dynamics / Solver Object.



In the same menu, create a gravity object and drag it into the solver object. As we mentioned before, all objects to be affected by the solver object must be a sub-object of the solver object. If an object is not a sub-object of the solver object will not be affected by the solver object. Next we need the objects that will collide with each other. Create a sphere, a cube and a plane (main menu / Objects / Primitive / Sphere, Cube and Plane). Assign the plane a width of 3000m and a depth of 1000m, the sphere a radius of 80m. **Set the number of segments for all three objects to 7 on each axis.** Leave the number of segments for the sphere set to "24" and set the other objects' segments to "7".



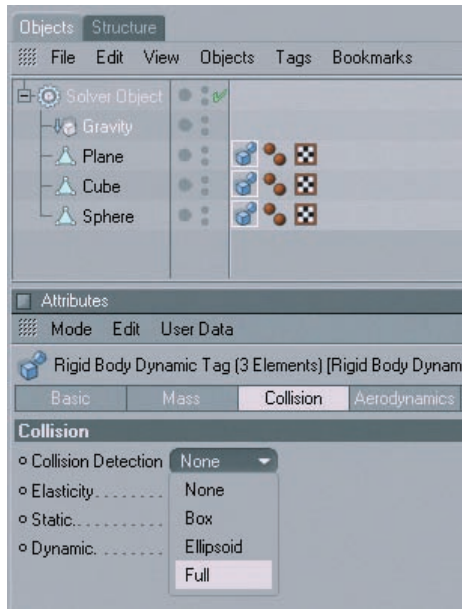
Zoom out so you can see the entire scene. Tilt the plane object in the “B” angle by 15 degrees and place the other objects in the approximate position they are in, in the following image.



The cube and the sphere should be placed quite a bit above the plane, each over one half of the plane as seen from the top so they don't cross paths. Drag the sphere, the cube and the plane into the solver object and convert them to polygons (Main Menu / Functions / Make Editable) or click on



The next step is the assigning of Dynamics properties using the Dynamics tags. For soft surface properties we would need to use Soft Body Spring Tags. Since we want to let the sphere and the cube fall onto a surface without them bouncing off, we need the Rigid Body Dynamic Tag. Apply to each object a Rigid Body Tag (right mouse button / Dynamics Tags / Rigid Body Dynamic). If you play the animation now, you will notice that all objects fall out of the picture. Of course this is not exactly what we want to achieve for this scene. This happens because all objects possess a mass. We need at least one object that does not leave its position so the other objects can collide with it. This is a job for the plane. In the dialog field, give the plane a “Total Mass” of “0”. When the scene is played now the plane object doesn't move but we're still not happy with the total result. The objects fall through the plane! The reason is that no object has a collision property. Give each of the three objects a “Collision” value of “Full”. To speed up your workflow you can select all tags at once and make the change to all objects. (You could also use the settings “Box” and “Ellipsoid” – these are significantly faster than “Full” and are often sufficient for simple objects).



Before we play the animation again we will change its length. To do this click on edit / project settings and change the “Maximum” parameter to 300 frames. So that our solver object doesn’t stop the animation at frame 75 we will also change its “Stop” parameter to 300 in the dialog field “Main” to the right.

The animation is still pretty boring but we can at least demonstrate what is expected of Dynamics: Two bodies fall and collide! Two further settings are still necessary to make the objects behave the way we want them to. Click on the solver object and set the value “Energy Loss” from 5% to 2%. This keeps the objects from being slowed too drastically. Now set the “Strength” value in the Gravitation objects’s “Field” dialog box to 3.

OK, here we go. Click on the play button and we’ll be finished with this chapter. If the objects are not behaving the way they should it could be due to the method of integration of the solver object and its oversampling setting. The higher the oversampling value the more exact the simulation of the solver object. Of course the time needed to calculate the simulation will increase as well.

Important Tip: If one of the objects is not positioned properly you can only change its position once the solver object has been deactivated. After repositioning the object you must select “Dynamics / Initialize Object” or “Initialize All Objects” in the main menu! Then the solver object can be reactivated.

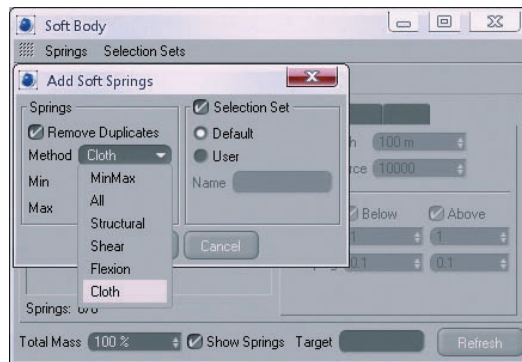
Try different property settings and see what changes to the Dynamic tags, gravity object and solver object to. Changes and variations are endless and need to be fine-tuned a lot until the result you want is achieved. Now let’s move to the second part of this Quickstart Tutorial: The soft bodies!

5. Quick Tutorial – Soft Bodies

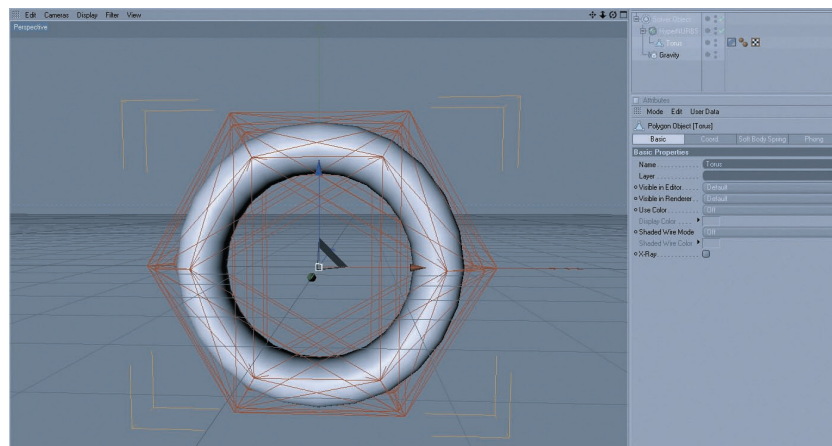
In this short chapter we would like to introduce Dynamic's soft bodies. A soft body object can be made up of an arbitrary number of polygon bodies. For our purposes we will use a primitive.

Open a new (empty) scene (File / New) and create a torus (Objects / Primitive / Torus). Rotate the "P" angle of the object by 90 degrees so the ring sits vertically and change the number of "Ring Segments" and "Pipe Segments" each to 6. Create a HyperNURBS object (Objects / Nurbs / HyperNURBS) and make the torus its sub-object. Don't forget to convert the torus to a polygon object (Function / Make Editable).

Create a solver object and a gravity object (Main menu: Dynamics / Solver Object, Gravity). Drag the HyperNURBS object with the torus and the gravity object into the solver object. Assign a Soft Body Spring to the torus (right mouse / Dynamics Tags / Soft Body Spring). A new window will open automatically in which you can change the parameters for the soft body tag. In the main menu of this window, select Springs / Add Soft Springs and set the "Method" in the subsequent window to "Cloth". Click "OK".



In the "Clothing" menu check the "Relax" box and close this window. If you've done everything correctly your editor view should reflect what you see on the following image.



If we play the animation now we will have a similar situation to the one we had in the previous chapter: The polygon object will fall out of the picture. We must set a zero value for the mass. But where? We will use the “Set Soft Mass” from the “Dynamics” menu. We will set the mass value of a point on the torus to “0” which will keep this point in place (since it will possess no virtual weight) and the rest of the torus will follow the laws of gravity.

To do this, select one or more of the top points of the torus in the point mode and select Dynamics / set soft mass. Change the mass value to 0 and click on “OK”.

Finally, change the integration method of the solver object from “Adaptive” to “Softbody” and start the animation again.

We’re done! The torus has a virtual hook and swings to and fro when you push it.

There’s not much we can say about the state of the physics of the polygon body but we can say a little about its physical state. The torus has assumed the properties of a rubber-like mass and will behave according to gravity and the properties we gave it. In the second part of this tutorial we only showed one of the many possibilities soft body makes possible. The variations are virtually limitless. Now we’ll come to the part that will improve your workflow.

6. Tips & Tricks

Always set the editor’s frame rate to “All Frames” (Animation / Frame Rate / All Frames). This function makes sure that every image of the Dynamics simulation will be visible and none will be suppressed since CINEMA 4D tries to adhere to the predefined amount of frames.

Depending on their complexity, Dynamics simulations can take very long to calculate and can place high demands on our CPUs. There are a couple of methods to achieve the same result without forgoing the smooth depiction in the editor window. A smooth depiction can save a lot of time in the “testing phase”.

It’s better if you use less-complex polygon objects and define them as sub-objects of a HyperNURBS object. In most cases the result will remain the same. The simulation of the soft body springs, for example, will be much more fluid since only a few springs will be used in the Dynamics simulation.

Bake the Dynamics simulation. This will transform the simulation into keyframes which has the major advantage that it will be depicted much smoother in your editor window and you can intervene and move or delete keyframes or even completely change a part of your simulation as you desire.

You can also use negative collision fields instead of collision. This method is calculated faster and delivers the same result in most cases. Try the settings “Ellipsoid” or “Box” and see how they change your simulation. These settings are faster and often deliver the same result here as well.

If your objects still pass through the plane simply increase the oversampling rate of the solver object. A value of 4, for example, means that Dynamics will calculate four additional images between each frame. An oversampling rate of 8 will result in twice the amount of images between frames, and so on. The higher the value, the higher the probability a collision will be calculated correctly. Of course the frame rate will suffer with each increase in oversampling.

As an alternative you can increase the Eps (Epsilon) value of the solver object. A higher value increases the area around the polygon object in which the collision algorithm occurs. By doing this the objects will be detected earlier and can be prevented from passing through the plane object.

We're at the end of this part of our Quickstart Tutorial. After all what you have learned in this tutorial we just want to say: If your simulation doesn't behave the way you want it to don't take it too "hard". You need time to master the forces of Dynamics, but after a short time of adjustment you will be able to master all of the physics forces of this powerful module and call yourself "Lord of the Masses"!.

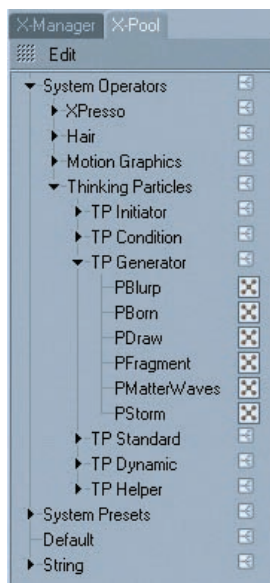
Welcome To Thinking Particles (Optional module)

This is the Quickstart Tutorial for Thinking Particles. It will give you a quick, and hopefully fun, introduction to this extensive particle system.

1. Introduction

Thinking Particles is a particle system. A particle system is used in the 3D world when a large amount of objects will be colliding such as a flock of birds, a bacterial invasion or (especially in conjunction with PyroCluster) fire and smoke effects. Let's say you have to animate a tornado sweeping through a desert. You would need to animate millions of tiny grains of sand flying through the air. If you had to do this manually you couldn't complain about job security because it would take you the rest of your life to do. Since you have other and more important things to do before you meet your maker we give you Thinking Particles. Maybe you are already familiar with CINEMA 4D's particle system which, in itself, lets you create amazing effects. Thinking Particles makes these possibilities endless.

Thinking Particles operates node-based. You can picture a node as a little machine which receives data at one end, processes it and outputs this processed data at the other end or transfers it to other nodes. If you are familiar with CINEMA 4D's expressions editor XPresso you will also have some experience with nodes. (If you have not yet worked with XPresso it would be a good idea to take a look at CINEMA 4D's XPresso Quickstart Tutorial since Thinking Particles requires some of the basic knowledge you will acquire there). Several elements of Thinking Particles are nodes – with the exception of the particle geometry object which we will examine in detail later. Thinking Particles uses an emitter node, not an emitter object as does the standard particle system. You construct your particle system like you do your XPresso setup and will have a comprehensive pool of nodes from which to choose, one of which is XPresso itself.



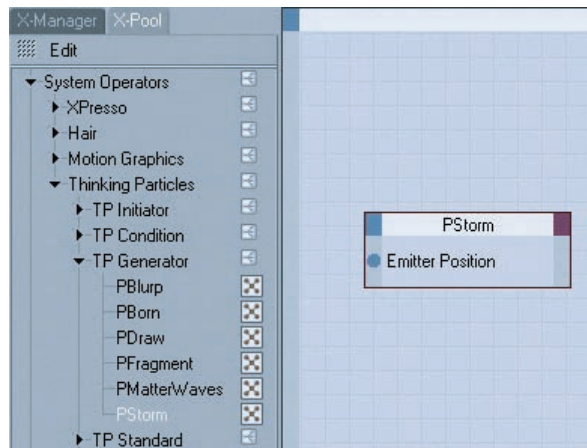
Each of these nodes has certain properties that can be changed using the Attribute Manager or can be connected to the ports of other nodes. This lets you affect particles in many different ways. The various groups can be subdivided, their look or behavior altered after they have been created and much more.

If you have worked with the “old” particle system up to now, Thinking Particles may seem a little complicated to you in the beginning. Complex can be good, though, since it opens a treasure chest of possibilities that let you implement your ideas! After all you have this Quickstart Tutorial at your disposal that will make learning Thinking Particles much easier. You will be shaking particle effects out of your sleeve in no time!

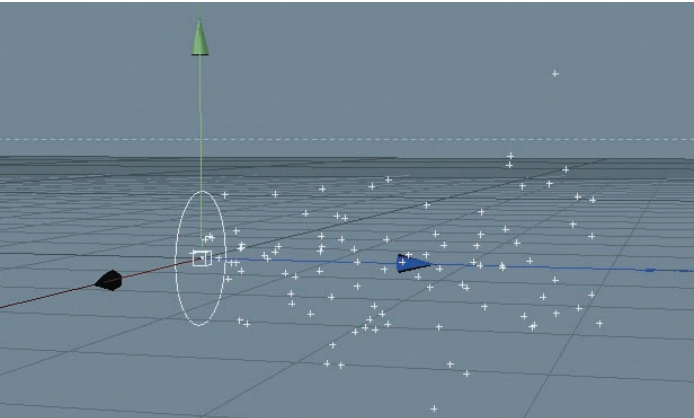
2. General Information / Interface

No emitter, no particles. The emitter is the source of all particles. As previously mentioned, a node serves as the emitter in Thinking Particles. In order to access this node we need to work in the XPresso editor. Create any object – a null object would be the best choice – and assign a XPresso Tag to it. The XPresso editor, our new playing field, will open. Here we will set up our particles.

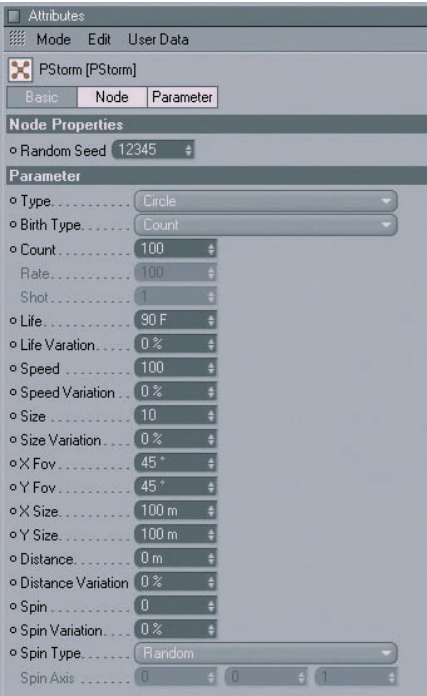
Several Thinking Particle nodes are listed in the X-Pool tab in the left column of the XPresso editor. Our emitter can be found in the “TP Generator” category and has the name “PStorm” (this is not the only emitter. “PSource” is a somewhat easier version). Drag the emitter into the field to the right.



We have already cleared the first hurdle and have created a particle emitter. You have probably noticed that you can now see a circle in the editor view. Play the animation and the circle will “spit” crosses – particles! These crosses won’t be visible in the renderer. They only serve the purpose of making the emitter’s behavior visible. We will find out in the following tutorial how to assign particles to “real” geometry.

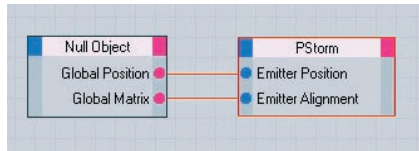


Select the emitter node in the XPresso editor and take a look at the Attribute Manager. Here you will see a bunch of values with which you can change the emitter’s behavior. How many particles should it generate? How long should the particles exist (live)? How big are they? Are they all the same size? You can change some of the values while the animation is playing and will see right away how the particles react to these changes.

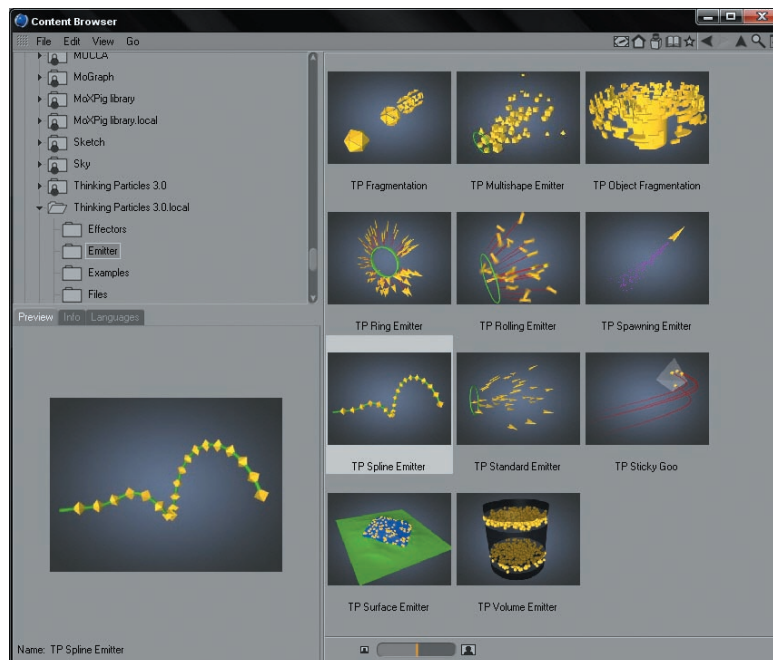


We still have a little problem, though. As you know the emitter is a node. A node cannot be moved, rotated or scaled. What do we do if we want to change the emitter's size or orientation? By simply assigning an object's coordinates to the emitter. Since we have already created a null object for the XPresso tag it stands to reason that we use it as a "donor" for the coordinates.

Drag the null object into the XPresso editor and assign it two output ports, one for "Global Position" and one for "Global Matrix". The emitter node already has an input for the emitter's position. Give it another input port – "Emitter Alignment". Connect both emitter's ports and we can already treat the emitter as an object. It will follow every movement and rotation the null object will make.



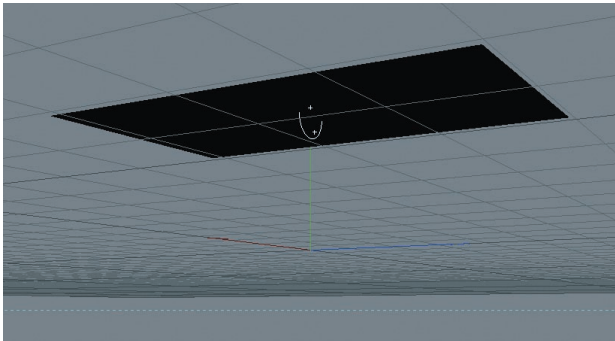
Even if it doesn't take a lot of time to create such an emitter it can be quite tedious repeating these steps for other emitters. This emitter has a critical disadvantage in that every parameter change requires the selection of the node in the XPresso editor. Never fear – MAXON has placed a standard emitter setup into the CINEMA 4D Content Browser's preset library that gives you convenient access to all emitter attributes over the null object's user data information. Here you can also make changes to geometry (standard is a cube) linked to particles with a simple drag & drop. The Content Browser's preset library also contains many more Thinking Particles setups. They're worth a try!



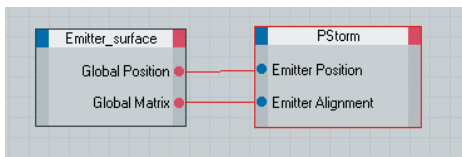
3. Quick Tutorial – Particle Snow

In this tutorial we will create a storybook snowflake scene. Of course we will need an emitter for this as well. This time we won't be attaching the emitter to a null object but to a polygon. We will use this polygon to control the size of the emitter.

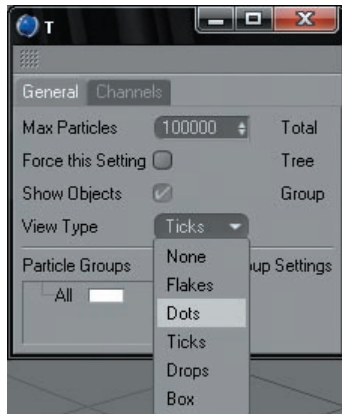
Open a new (empty) file. Create a square polygon primitive (Objects / Primitive / Polygon). Convert the parametric object to an editable polygon (Functions / Make Editable). Name the object "Emitter_surface" and position it to Y=300 (snow most often comes from above). Scale it to X=600 and Z=1000. The snow flakes will be generated from this surface.



Assign an XPresso expression to the plane and the XPresso editor will open automatically. Drag the "Emitter_surface" onto the XPresso editor window. Create a "PStorm" node. Again the nodes must be connected to each other so that the polygon's coordinates can be transferred to the emitter.



Now our emitter is busily emitting little crosses. By the way, the particle preview doesn't always have to be depicted as crosses. Take a look at the Thinking Particles settings (Objects / Thinking Particles / Thinking Particles Settings). Here you will find a menu you can use to change the particle's look, to that of flakes or points for example.



The particles should be created on the surface of the polygon. In order to do this the setup has to be extended a little. Give the “Pstorm” node a “Particle Birth” output. This output contains various particles generated by the emitter that will be “processed further”.

Next we need a “PSurfacePosition” node (XPool / System Operators / Thinking Particles / TP Helpers). This node creates random positions on a given polygon object. It has an “object” text field in its Attribute Manager. Drag the emitter surface into this field. You will also find a menu in the Attribute Manager that determines if the particles will be emitted from the face, edges or points of the polygon. Leave the setting on “Face” because we don’t want the particles to be emitted only from the corners.

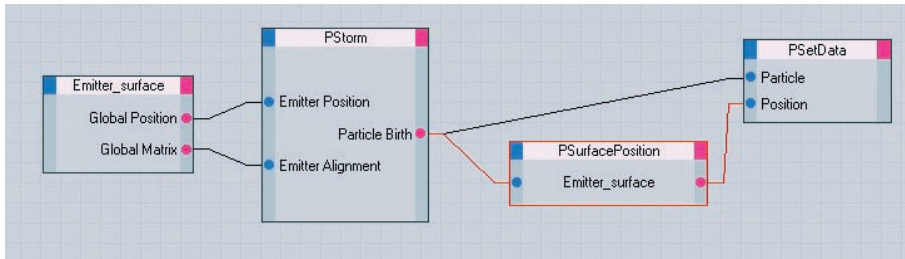


This node lets you arbitrarily select a polygon object as an emitter. It’s only important that the parametric object is first made editable. Just for fun, try to generate snow emitting from a sphere.

We need another node which will tell the particles to orient themselves according to the “PSurfacePosition” node’s position. This is what the “PSetData” node is for (XPool / System Operators / Thinking Particles / TP Standard).

Drag the “PSetData” node into the XPresso editor and assign an input port for position. This will be the property we want to set for the particles. This port will be connected to the position output of the “PSurfacePosition” node. Connect the particle input port of the “PSetData” with the “Particle Birth” output of the “PStorm” node. Now “PSetData” knows which particles it will position. Now connect the “PStorm” node’s “Particle Birth” output port to the “PSurfacePosition” input node.

The first steps have been made! Lean back, take a deep breath and take a minute to admire the setup. Isn’t it beautiful?



To summarize:

The emitter “passes along” its generated particles to the “PSurfacePosition” node. This node, in turn, calculates the random positioning of the particles on the polygon and transfers this information to the “PDataGet” node.

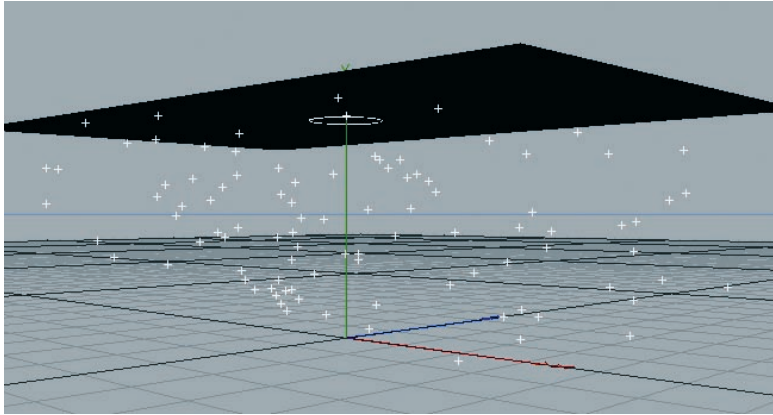
This node assumes the actual setting of the particles in their random positions

You can see the result immediately in the editor view. The particles are now being generated evenly from the entire polygon. The particles are still moving in the wrong direction – along the default Z-axis of the emitter. Since snow doesn’t usually fall horizontally (except in a major storm), we have to rotate the emitter by 90 degrees. Keep in mind that the emitter is an object and therefore cannot be transformed directly. Its orientation is controlled by the “emitter surface’s” coordinates.

Select the “Emitter_surface” and switch to the “Use Object Axis Tool”.



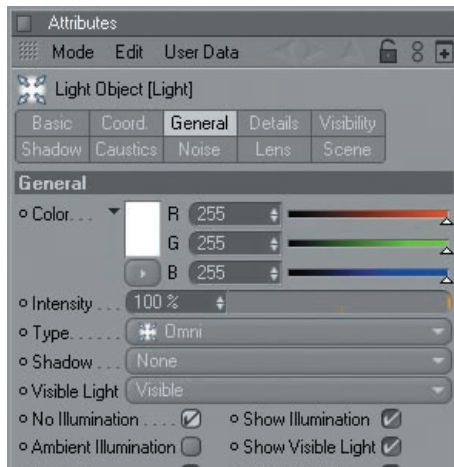
Rotate the object axis 90 degrees around the red X-axis so that the blue Z-axis points down. Presto! The snow is now falling down, as it should.



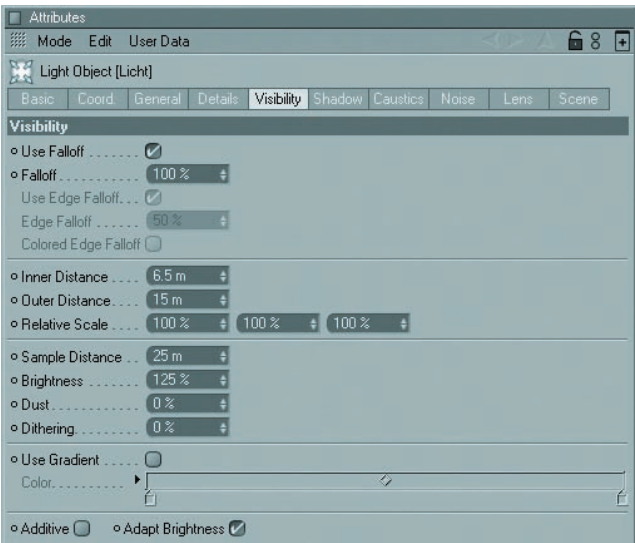
4. Quick Tutorial – Objects as Particles

Now we have a cute little setup that makes snow fall but we're still missing the actual snow. What this snow will look like is completely up to you. Simple spheres or snow crystals with alpha channels? We've decided on a light object. With the right settings you can make it look like a pretty convincing snow flake.

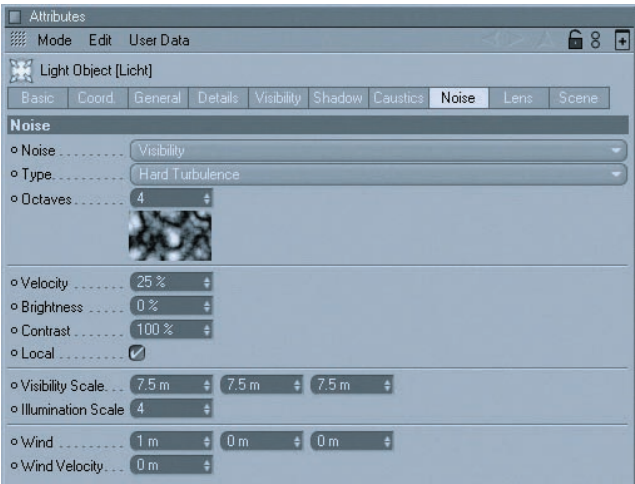
Add a light source to the scene (Objects / Scene / Light). You will have to change some of the light's settings in order to make it "flaky". Set "Visible Light" to "Visible" and "Noise" to "Visibility" on the Noise menu of the Attribute Manager. The light should not be visible and should not light the scene so activate "No Illumination" in the "General" tab.



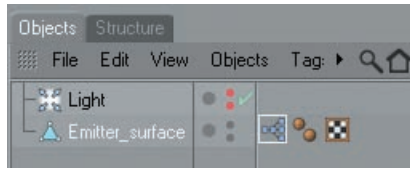
Switch to the “Visibility” menu and set the “Inner Distance” to 6.5, the “Outer Distance” to 15 and raise the “Brightness” to 125%.



Switch again to the “Noise” menu. Set the “Type” to “Hard Turbulence” and all three scale parameters to 7.5.

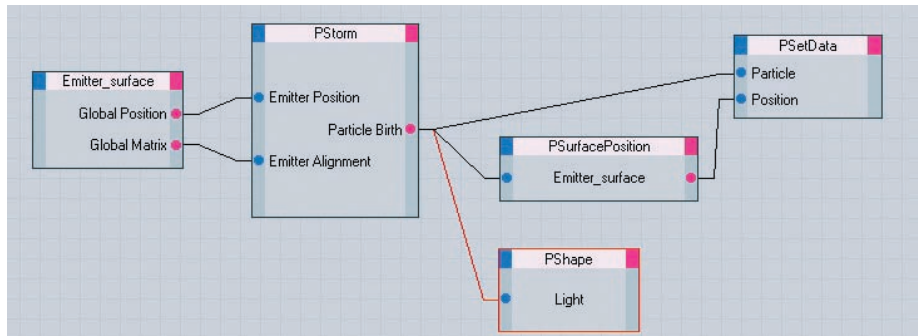


Make the light invisible for the editor and the renderer in the “Basic” menu. It will later be visible only as particles.



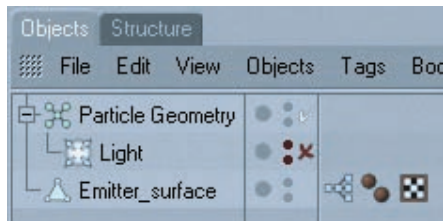
The “light flake” is ready. Next we have to let Thinking Particles know that it should use this light as a particle. You guessed it – there’s a special node for this as well. It’s called the “PShape” node and is located in the X-Pool tab in the “TPStandard” category.

Add a “PShape” node to the existing XPresso setup. Drag the light object into the object text field in the node’s Attribute Manager. Connect the input port of the “PShape” node to the output port of the “PStorm” node. Set the “Bounding Radius” to 15.

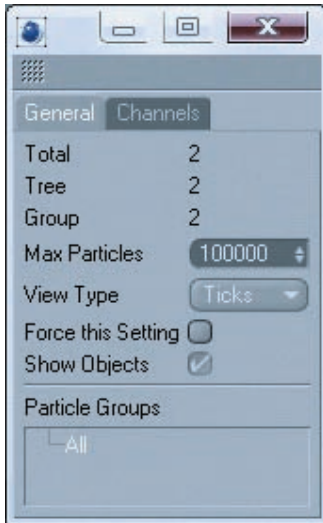


Just one more step. In order to be able to display a renderable object at all, Thinking Particles needs a special “Particle Geometry” object.

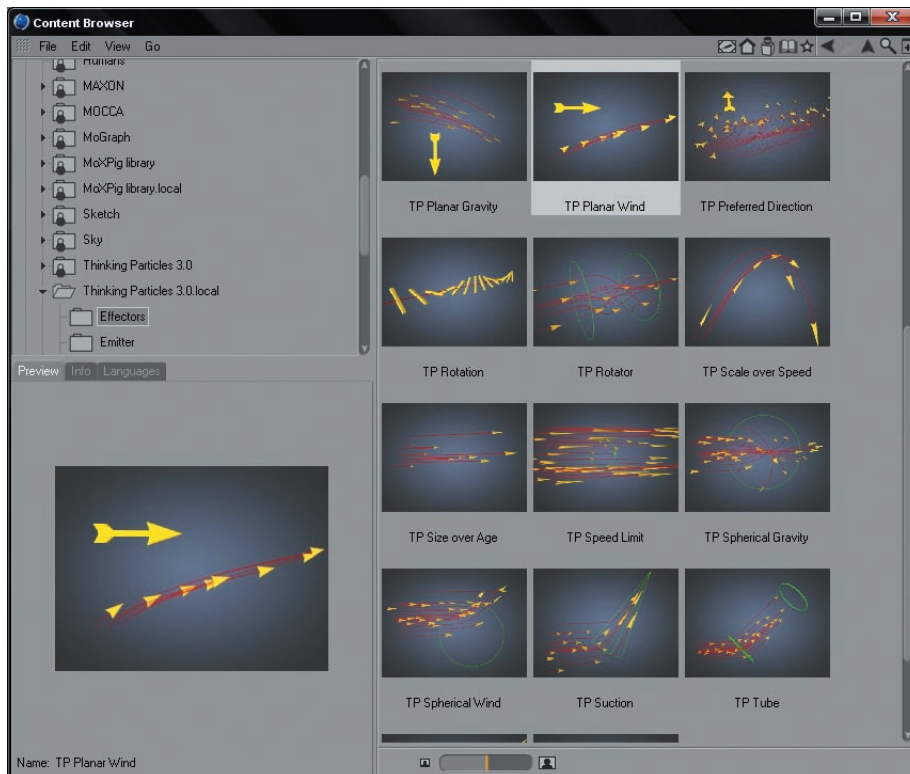
Create a “Particle Geometry” object (Objects / Thinking Particles / Particle Geometry) and drag the light from the Object Manager onto it so the light becomes its sub-object.



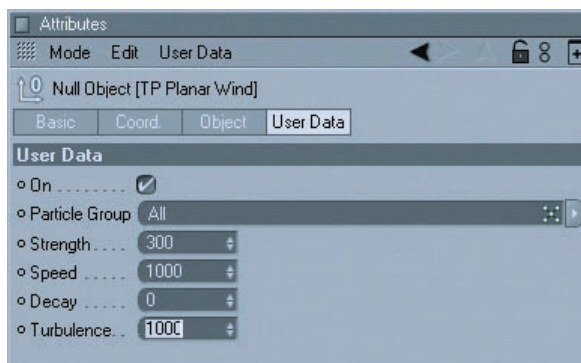
In order to understand the purpose of this object you have to know that Thinking Particles can divide particles into groups. To a certain extent, the “Particle Geometry” object serves to let you assign various geometric objects to these groups. Our example contains only one group named “All” which contains all particles that will be generated. Nevertheless, a “Particle Geometry” object is always required. Make it a habit to make the object – in this case the light – a subordinate of the particle geometry even if it’s not compellingly necessary for just a single particle group. All of a scene’s particle groups are listed in Objects / Thinking Particles / Thinking Particles Settings.



How about a little snowstorm to finish? The emitter node’s attributes offer us a wide variety of possibilities for changing the behavior of the flakes. We can shorten their life span, make them fall faster or let them rotate. It’s also possible to influence particles that have already been generated. Open the Content Browser’s preset library in the main menu and look for “TP Planar Wind”. As the name suggests, this option is a wind-like force which we can use to add a little turbulence to our snowfall.



Add “TP Planar Wind” to the scene. The wind does not yet know which particle group it should influence so open the Thinking Particle Settings (Objects / Thinking Particles / Thinking Particles Settings) and drag the group “All” into the “Particle Group” text field in the “User Data” menu of the “TP Planar Wind’s” Attribute Manager. Set the “Strength” to 300 and the “Turbulence” to 1000.



Our winter wonderland has been turned into a gyrating snowfall. We hope you had fun with Thinking Particles in spite of the snow storm at the end. Use this tutorial as a starting point and try new things with the snow. Let it fall on other objects or transform it to rain. The only limitation is your own creativity – which we know is limitless.



5. Tips & Tricks

Don't make your life more difficult than it has to be (which doesn't just apply to working with Thinking Particles) and make use of the object library where you will find the most important Thinking Particles Nodes already assigned to objects and with simple interfaces which, in most cases, will save you a lot of time.

Organize your setup. XPresso and / or Thinking Particles setups can become very complex. It helps to keep your setup organized as they are being created. Nodes can be grouped and good annotations will definitely help if you don't want to wonder what in the world you did when you created this scene three weeks ago.

Don't forget that particles can increase the size of your scene dramatically. If you create an emitter that generates 10,000 objects and 10,000 polygons in every frame even the fastest computer will drop to its knees (figuratively, of course). We recommend you work with the faster preview functions until the very end of your project before assigning particles to geometry.

6. In Closing

We have reached the end of our Thinking Particles tutorial. You have carefully tested the waters and we hope you stayed in. Maybe our little demonstration has encouraged you to move on to bigger and better things. Go ahead and experiment! Have fun! As we mentioned at the beginning, Thinking Particles can be very complicated but the most complex setup begins with a single node. If you take it step-by-step it will only be half as bad as you think ...!

On that note, we hope this little tutorial has helped you on your way to a (particle) stormy future!



Welcome To MOCCA (Optional module)

This is the Quickstart Tutorial for the MOCCA module, a collection of tools especially for character animation.

1. Introduction

3D character animation is a complex subject. It not only challenges the software but the animator as well. Almost anyone can quickly create a character that moves in one way or another, but a convincing character animation requires quite a bit more work. The animator needs to know the theory behind character animation before he or she can judge whether or not the animation is good or bad.

So before we pounce on the MOCCA palette we would like to quickly go over the **12 principles of character animation**. These 12 rules were developed back in the 1930s by Disney animators and can almost be applied 1:1 to modern 3D animation. They are useful not only for characters but for just about anything you animate. Anyone who is serious about character animation should stick these rules to the ceiling over their bed (and get rid of whatever else that is up there now ;o). A convincingly animated character will then be only a matter of patience.

Squash & Stretch – every organic object deforms in some manner when it is in motion. Squash and stretch is the exaggeration of this effect when a character is in motion.

Staging – is the setting up of an animation. This includes lighting, camera angle, effects and slow motion.

Anticipation – Expectation of a reaction to an action is known as anticipation.

Pose to Pose / Straight Ahead – these terms describe two animation techniques. Pose to Pose sets two important key poses and the time between is then “filled”. Straight Ahead sets one key after the next – straight ahead through time.

Follow Through / Overlapping – the impetus or the overlapping of movement of a body parts.

Ease In / Ease Out – an object begins to move slowly, reaches its top speed and slowly comes to a halt.

Curves – In nature almost every movement is described as curves, no matter if it's a swinging arm or the rotation of a head.

Secondary Action – movement that is created by movement of another object.

Timing – the speed with which an action is executed.

Exaggeration – can be used in many instances: Arms that are being stretched too long when a character is hanging from a cliff, eyes jumping out of their sockets to help demonstrate a shocked expression.

Appeal – a character's impression, look and nature.

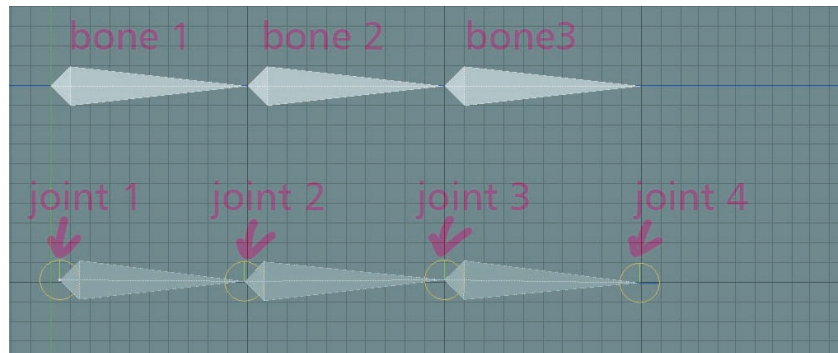
Personality – like an actor, a good animator must be able to make his character express himself.

You can find more detailed descriptions of these rules online or in any good animation book. These rules might seem pretty extensive at first glance but the more you animate the more they will become your own flesh and blood. Every time your animation “looks a little weird” it will most likely be because you didn’t follow one or more of these rules. Now we want to put these rules to use.

2. General Information/Interface (Joints and Rigging)

If you are a R9.x user and have just upgraded to R11, here are a few infos regarding changes since Release 9.x. Version 3 of MOCCA features a completely reworked system since R9.x where you now work with “joints” instead of bones. Joints have been a part of the main CINEMA 4D application since R10 and bones now exist only for compatibility with old scenes. MOCCA 3 itself offers enhanced tools designed to help you create perfect character animation fast.

Joints are laid out differently from bones because they represent the joints between bones, not the bones themselves. Where you would have used three bones in R9.x, four joints are now required, as shown in the next screenshot:



The character is no longer directly deformed by the bones/joints but by a Skin object that is a child of the polygon mesh. Among other things, MOCCA now contains a Bind function. Select the joints and polygon mesh you want to be deformed and the Bind function creates the Skin object automatically as a child of your mesh and adds a Weight tag to the polygon mesh.

Your character is weighted automatically and in most cases the weighting should be adequate. The time-consuming task of painting the weights manually using the Claude Bonet tool is no longer necessary!

You can quickly correct any areas of the deformation that need fine-tuning using the Weight tool. Also, unlike bones in R9, joints do not have to be children of their polygon object. They are independent of hierarchies, allowing you to deform multiple characters at the same time using a single skeleton of joints.

Since Version 3 MOCCA has a the Muscle object and Muscle deformer. These make it possible for you to place virtual muscles under your character's skin that bulge when the joint bends, causing the surrounding mesh to deform.

MOCCA also includes three very powerful tools: Morph, Vamp and Visual Selector.

Like the PoseMixer tool, the Morph tag lets you create various facial expressions for your characters and morph between them. The difference is that with Morph, you no longer need to work with copies of the original mesh, as was required in Posemixer for R9. Your polygon object acts as the reference and a "base morph" (starting position for all following morphs) and "target morph" are created in the Morph tag. You select the morph target in the Morph tag and change the mesh...finished!

You create another morph target for each additional pose and model the poses one after the other. All the expressions are stored in a single tag. Also, when using the Morph tool, there's no need to worry if you have to make changes to the mesh after creating the poses. The poses will still work! Suppose you've created all the poses for your character, but decide it would look much better with a second nose. The Morph tool will still happily morph between the poses.

Vamp gives you the possibility to transfer data from object to another, including selection information, Texture tags, vertex maps and UVs. You can even transfer facial poses from one character to another!

Visual Selector is a great help with day-to-day animation. You load a render of your character into Visual Selector's background (or use Visual Selector's default character picture) and place your character's controllers onto the picture in the appropriate places. Visual Selector removes the need to keep looking for your character's controls in the Object hierarchy. Everything is now represented visually and you can, for example, select the foot controller by clicking on it directly in the picture. You want to move the eyes? No problem. Click on the controller for the eyes directly in the picture.

You'll find the MOCCA commands in the main menu under "Character", or you may prefer to integrate the MOCCA toolbar into the layout (as described for the Dynamics toolbar in the "Dynamics" chapter).

If you're new to the process of rigging characters, the following overview may help.

As with a real human, your character needs a skeleton of bones (or in our case, joints) in order to be able to move around in the world. You place the joints inside the character's mesh. The joints are linked to the mesh via a Weight tag and Skin deformer so that each joint knows which part of the geometry to affect.

You can weight joints by selecting them and painting directly onto the mesh using the Weight tool. While the Weight tool is active, the mesh is displayed black and the currently painted weighting is shown in white. The joint now knows it should affect the white painted parts of the mesh only. In the active Weight tool mode, weighting is shown for the selected joints. Each joint has its own weighting.

The joints must be arranged into a hierarchy in the Object Manager in a similar structure to the bones in your own body. In real life, when you move your upper arm, the lower arm and hand move with it because they are effectively children of the upper arm.

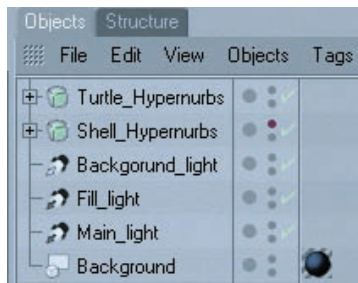
Likewise, in CINEMA 4D's Object Manager, the elbow and wrist joints must be children of the shoulder joint. If you move the shoulder joint, the child joints will move with it together with the mesh weighted to the child joints — even though the shoulder joint is weighted to the upper arm only.

As previously mentioned, each child joint has its own weighting and moves the parts of the mesh not weighted to the shoulder joint.

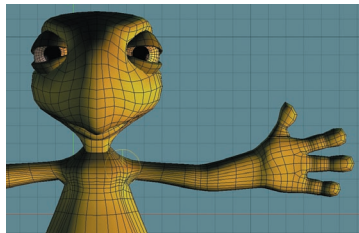
Don't worry if this seems complex. All will be explained in the rest of this chapter.

Open the file "QS_Joints_Start.c4d". Although turtles are generally considered to be slow creatures, this one will help us to get a fast result. In this tutorial, we'll add joints to the left arm so that we can move it.

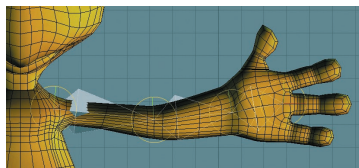
You should already have a good view of the left arm in the editor, ready for joints to be added. First we need to hide the turtle's shell so that it won't be in the way later on when we weight the body. To hide the shell, click twice on the top grey dot in the Object Manager to the right of the "Shell_Hypernurbs" object. The dot should turn red and the shell should now be hidden.



We can add joints using the Joint tool. Select it by choosing Character/Joint Tool from the main menu. In the editor, Ctrl-click on the turtle's shoulder to add the first joint.



Add the other joints required for the arm by CTRL-clicking on the elbow, wrist and base of the middle finger, as shown in the next screenshot.

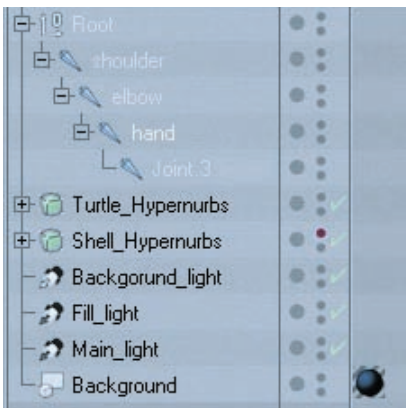


The joints are shown as yellow circles in the editor and as objects in the Object Manager. (When adding joints, make sure that the “shoulder”, “elbow” and “hand” joints don’t form a straight line. The “elbow” joint should be placed slightly to the rear, as seen from the top view, so the IK Chain tool (discussed later) will know in which direction the arm should bend).

Switch to the top view in the editor’s menu (Cameras/Top) and zoom the window as needed to get a good view of all four joints. As you can see in the top view, the joints are a little behind the geometry. It’s important to get the joints in exactly the right place inside the geometry in order for the automatic weighting to work correctly later on.

When we created the joints in the front view, the Joint tool had no way to tell at what depth the joints should be placed. We can deal with this now. The top view gives us the ideal place to move the joints inside the geometry. In the Object Manager, click on the “Root” object (top object in the hierarchy of joints) and in the editor, drag it a little along its negative Z-axis (downwards in our editor) until the joints are inside the geometry.

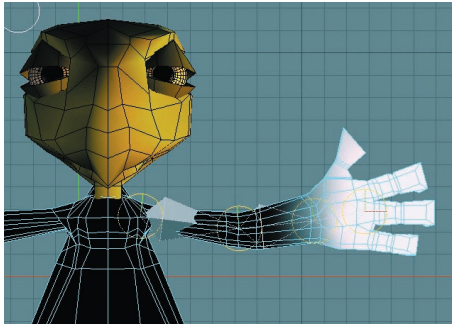
Switch back to the front view. Zoom the window as needed to get a good view of the arm. Rename the joints from “Joint”, “Joint.1” and “Joint.2” to “shoulder”, “elbow” and “hand” respectively (double-click).



In the Object Manager, click on the “+” symbol for “Turtle_Hypernurbs” to open the hierarchy for the turtle’s mesh. Hold down the Ctrl key and click to select the “body” object and the three joints “shoulder”, “elbow” and “hand”. Make sure that only these four objects are selected and choose Character/Bind in the main menu.

A Skin object is added automatically to our mesh as a child and a Weight tag is added. The Skin object will take care of the deformation as the joints are moved and the Weight tag stores the weighting created automatically by the Bind function for the individual joints.

Let’s take a look at the weighting. Deactivate the “Turtle_Hypernurbs” object in the Object Manager (the white check mark turns into a red cross). Click on the “hand” joint and choose Character/Weight Tool in the main menu. You’ll see the joint’s weighting appear in white in the editor. Black geometry means no weighting.



Things are looking good so far. If you've painted weights in previous versions of CINEMA 4D using the Claude Bonet tool, you'll appreciate just how much effort the automatic weighting has saved us here. However, if you now select the shoulder joint, there appears to be problem: almost all the geometry is weighted to this joint. Also, the elbow joint has more geometry weighted to it than it should have. Both these joints are affecting parts of the geometry they shouldn't.

But don't worry. The auto weighting is in fact working correctly. The reason is that we didn't create a joint for every part of the body. When we called the Bind function, it tried to auto weight the entire geometry using just the four joints.

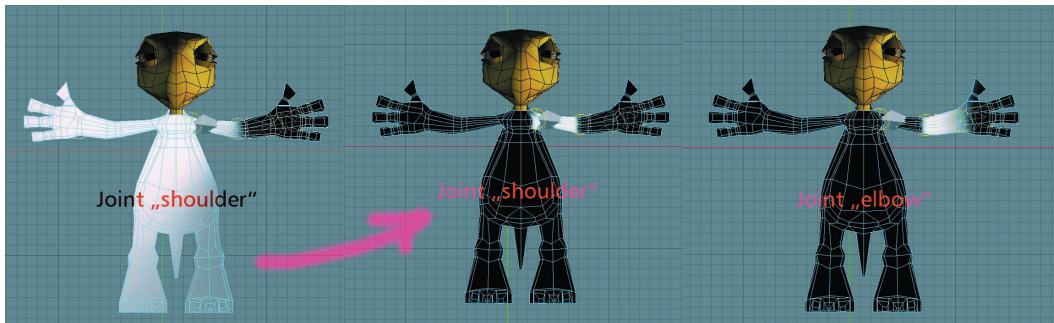
When animating characters you usually create a joint for each part of the body. After all, even if you just wanted to wave an arm in the air you'd still expect some movement to come from the rest of the body. In most cases, you'll want the function to weight the entire mesh. In the case of our simple arm, that's not what we want but it gives us an opportunity to take a closer look at the Weight tool.

Like the old Claude Bonet tool, the Weight tool lets us paint weights onto our character's geometry.

In the Object Manager, select the "shoulder" joint and choose Character/Weight Tool in the main menu. If you deactivate the HyperNURBS, you'll see the weighting appear in white and you can paint to add or remove weighting. When you select the Weight tool, its settings appear in the Attribute Manager at the bottom right.

Disable the "Visible Only" option and with Ctrl held down paint over the turtle's white weighting in the editor. Disabling this option lets you also paint weighting onto surfaces that are behind other surfaces and in real life would be blocked from view.

Painting on black areas adds weighting. Painting with Ctrl held down subtracts weighting. Use the Weight tool to adjust the weighting as shown in the next screenshot, paying close attention to the neck (you may find it helpful to hide the head to get a better view of the neck).



Once you’ve adjusted the weighting, activate the HyperNURBS (“Turtle_Hypernurbs”). Make sure the Model tool is selected



and click on the Rotate tool.



If the “shoulder” joint is selected, you should now be able to move the arm around. If parts other than the arm also move then these parts have weighting that needs to be removed. Use the Weight tool as before to remove the weighting from these parts. Also try rotating the “elbow” and “hand” joints.

Congratulations! You have just weighted the arm. The arm is now ready to be animated using forward kinematics, which we’ll move right on to in the next chapter.

3. Quick Tutorial – Forward Kinematics / Inverse Kinematics (FK/IK)

Forward kinematics (FK) and inverse kinematics (IK) — terms used in 3D animation — give you two different ways to animate. FK means rotating every bone within the hierarchy from top to bottom until the arm is in the desired position. This method has its advantages for some parts of the animation but disadvantages in other parts.

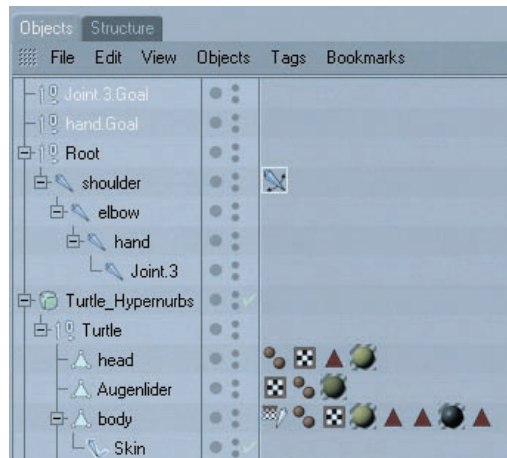
It can be difficult to tell how much you need to rotate, say, the upper arm then lower arm then wrist for the hand to reach a very specific position. Or suppose you’ve worked your way down to the hand and then notice that the upper arm is raised too high. You rotate the upper arm to correct this, but this rotates the children as well, putting the lower arm and hand out of position. These children must be rotated again to be put back into position.

With these difficulties in mind, what if you could simply grab the hand and move it to where you want it with the rest of the arm moving itself to follow the hand? This is what IK allows you to do. Instead of rotating the individual joints one by one and working down the chain, you have an “effector” located near the hand that you can move to control the movement of the entire IK chain in one go.

In this chapter, we’ll create an IK chain for our turtle from the previous chapter so that we can then animate it using IK. Open the file “QS_Joints_Final.c4d” or, if you’re carrying on from the previous tutorial, feel free to use the result of that tutorial as your starting point.

Another key feature in MOCCA 3 is the super fast IK setup. You select the joints, call the IK Chain tool and that’s it — the arm is ready to animate using IK! You no longer need to go through a long process of setting targets, constraints and so on. In the case of the arm, it’s ready to animate in just two steps.

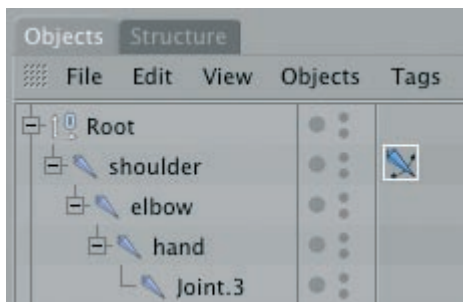
Now it’s time to get started. Hold down the Ctrl key and select the joints “shoulder”, “hand” and “Joint.3”. Hold down the Ctrl key again and choose Character/IK Chain in the main menu while pressing the CTRL-key. That’s it for now! The Object Manager should look as shown in the next screenshot.



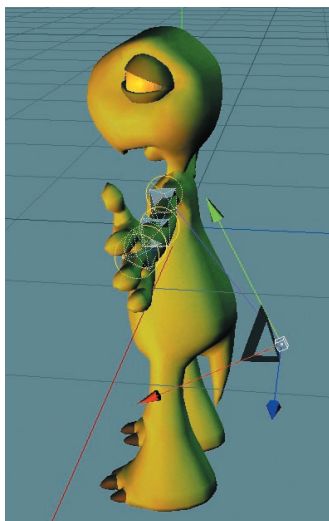
Before we carry on working with the arm, we should switch to the perspective view so that we can see the arm in three dimensions (Cameras/Perspective).

You’ll find two new objects at the top in the Object Manager: “Joint.3.Goal” and “hand.Goal”. Select both objects (Ctrl) and make sure that the Model tool is selected in the left toolbar. Select the Move tool in the top toolbar. Zoom the window as needed to get a good view of the arm and move the object axis around in the editor. The hand moves around and the rest of the arm follows it. It is not even necessary to set the Pole Vector to make the elbow bend correctly. However, in order to have control over the elbow we will still set a Pole Vector ().

To do this, select the “shoulder” joint’s IK tag



and in the Attribute Manager (bottom right), click on “Add Pole”. You’ll see a new object named “shoulder.Pole” appear in the Object Manager on the same hierarchy level as the “shoulder” joint. Select the pole and in the editor, drag the pole slightly down and behind the turtle.



Select the two effectors again (“Joint.3.Goal” and “hand.Goal”). Move the arm forwards and backwards again. Our arm behaves as before but now we have complete control of the elbow (and all arm rotations) thanks to the Pole Vector.

Two more quick tips:

You will surely have noticed that the entire arm rotates when the Pole Vector is pulled and the palm of the hand (depending on where you have placed it) no longer points forward. It would be better to place the Pole Vector behind the elbow instead of slightly below it. We will change this now in just a few steps. Ideally you should always set up a character’s complete rig before you call up the “Bind” function that connects the mesh with the “skeleton”.

No part of the polygon mesh should be moved after the “Skin Object” is turned off. This means that the activation of the Skin Object itself constitutes a deformation. However, we want the deformation of the polygon mesh to be caused by the character’s movement. Try turning the Skin Object off. The mesh will jump away from the joints. This can, however, be rectified in just a couple of steps:

Leave the Skin Object turned off and move the Pole Vector back up (Y-direction) until it is at the same height as the elbow (as viewed from the front). You will see that the joints will abandon their position and that the polygon mesh will ignore this altogether. Unfortunately our joints no longer have the correct position and protrude through the mesh.

This can be rectified by selecting the “Joint.Object.3.Target” and “Hand.Target” objects and moving them slightly forward and down. Use the perspective view to make sure that the joints are back within the mesh. Now click on the “Weight” tag to the right of the “body” mesh in the Object Manager and go to the “Joints” tab. Select all three joints and select the “Tag” tab. Click on “Set Pose” and you’re done. Now our mesh no longer jumps when the Skin Object is turned on and our Pole Vector is where we want it - directly below the “elbow joint”.

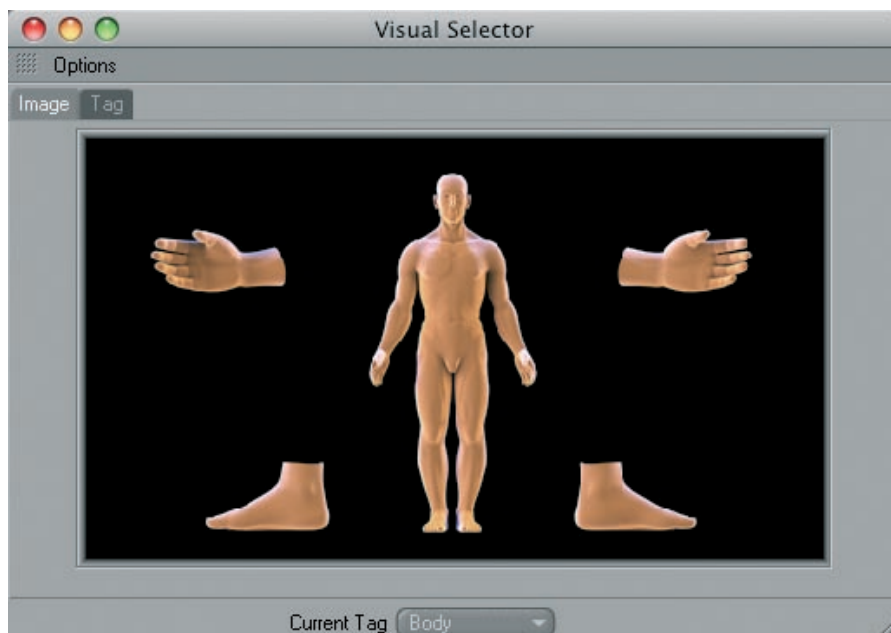
Although joints in R11 are generally faster than in R10.x, you can speed things up further in the editor by setting the Skin object’s “Type” (Object tab) to “Linear” (remember the Skin object from before? It’s a child object of the deforming mesh that takes over the deformation for the joints). The “Linear” deformation is calculated faster than the “Spherical” method, but the results are not as good. “Spherical” achieves better results by preventing the mesh from shrinking. Simply switch the “Type” back to “Spherical” before rendering your animation.

4. Quick Tutorial – Visual Selector

Next, we’ll take a look at the Visual Selector, which is designed to save you time when selecting your character’s controller objects. Suppose you’ve fully rigged your character and you’re ready to animate. At the top in the Object Manager, you have your hand and foot targets, and nested deep within the hierarchy are the Morph tags, Pole Vectors, hip and head joints, target for eyes, and so on. Finding the various controls can slow you down substantially while you are animating. As a character animator, you’ll want to concentrate on the essentials so that you can quickly take the next step without having to rummage around for the tools you need.

This is where Visual Selector can help. With Visual Selector, you only need to search for the controllers once: when you create your Visual Selector interface.

Start a new scene (File/New) and add a cube (Objects/Primitive/Cube). In principle, it doesn’t matter which object you choose because the Visual Selector tag can go on any object. In the Object Manager, right-click on the cube and choose Character Tags/Visual Selector to add a Visual Selector tag to the cube. The Visual Selector interface opens automatically. Here we can see the default Visual Selector background with a character in front in a typical character setup pose with enlarged hands and feet.

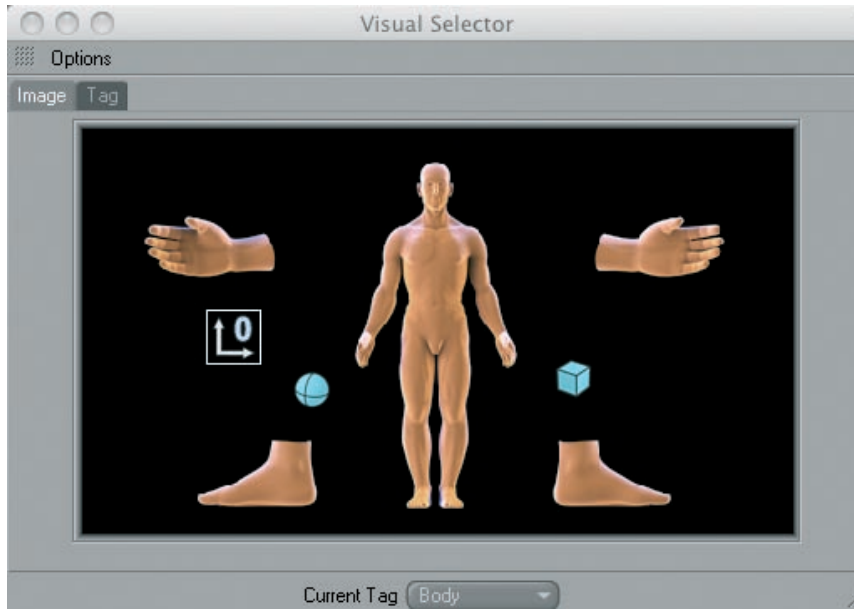


In the following text, the abbreviation “VS” for “Visual Selector” is used.

The purpose of VS is to place the individual controllers required to animate the character onto the corresponding parts of the picture. For example, a right hand controller should be placed on the right hand in the picture.

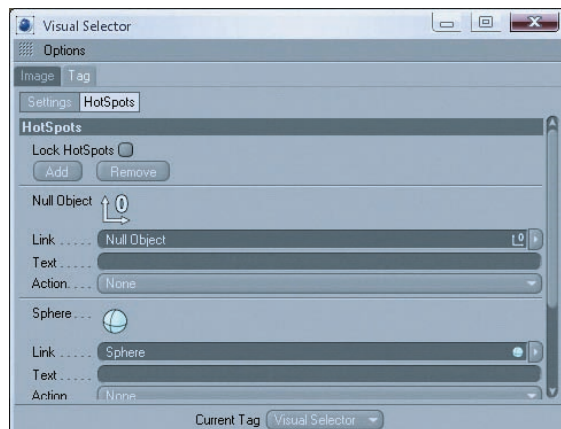
Best of all, you can drag and drop the controllers directly from the Object Manager onto the picture in the VS interface. Virtually any type of object can be dragged into VS. You can leave the visual representation as it is or assign a different icon or color to the object.

Create a cube, a sphere (main menu/Objects/Primitive/Cube and Sphere) and a null object (main menu/Objects/Null Object) and drag them one by one from the Object Manager into the VS interface.



The object icons appear as a layer on top of the character image. Drag the icons around on the character image to wherever you want them placed. Move the mouse pointer over one of the icons and click and hold down the right mouse button. You now have two possibilities. Either press the left or right cursor key to change the icon, or press the Up cursor key to open a window containing all the available icons, where you can choose which one to use. The color field (currently white) in the left part of the window lets you change the icon's color.

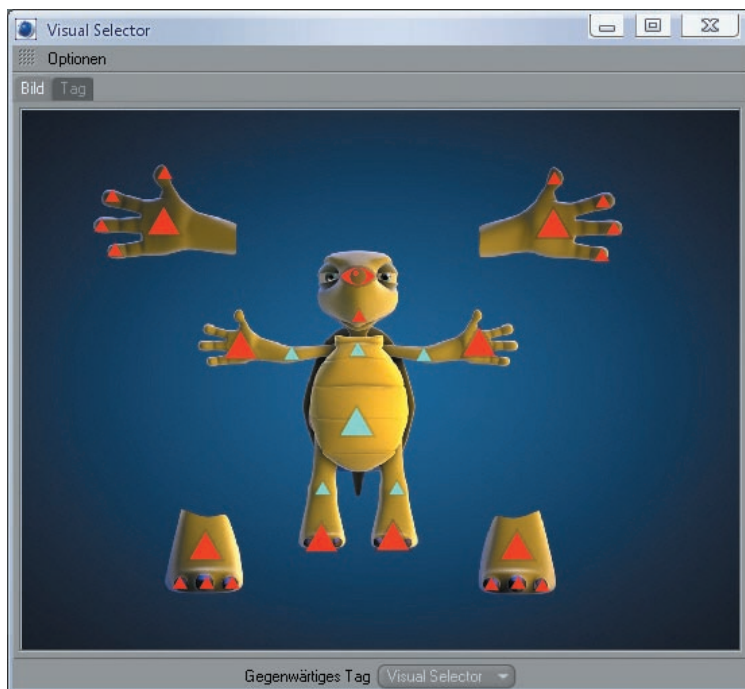
At the top in the main VS interface you'll find a Tag tab. Click this tab to access the VS settings. Here you can give a name to the VS, swap the background image for your own rendering and change the aspect ratio and size of the VS. On the Tag tab you'll also find a "hotspots" page. Clicking on this takes us into the settings for the hotspots you've created.



Here you can change the link to the hotspots and enter a name for the hotspot that appears in the main VS image window next to the hotspot. A further highlight is the Action function, which lets you assign a function to the hotspot (another feature designed to save you time).

Suppose your hotspot is linked to the head joint, which must be allowed to rotate but not moved or scaled. Simply assign it the Rotate action. Now, each time you select this hotspot, the Rotate tool will be activated automatically no matter which tool was previously selected.

Now you know the basic principles of using VS. Usually, you'll use your character's controllers, Morph tags and so on as hotspots rather than the cubes and spheres we used in this quick tutorial. Below you can see a typical VS setup.

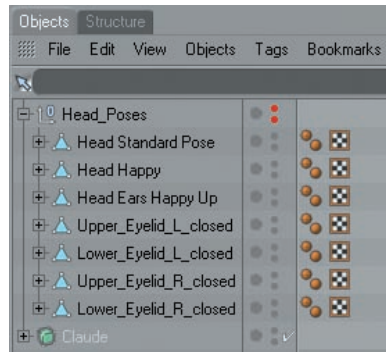


5. Quick Tutorial – PoseMixer

As we have seen, joints are good for constructing a “control frame” for a figure. You can also use joints to animate the face. The joints are positioned so they can imitate the natural facial movements. A second possibility is to “morph” the facial geometry using different states. For this MOCCA offers two tools: the PoseMixer, which we will examine closer in this chapter, and the Morph tool, which we’ll return to later. The PoseMixer differs from the Morph tool in that it also works with bone hierarchies. For example, you can “pose-mix” finger positions.

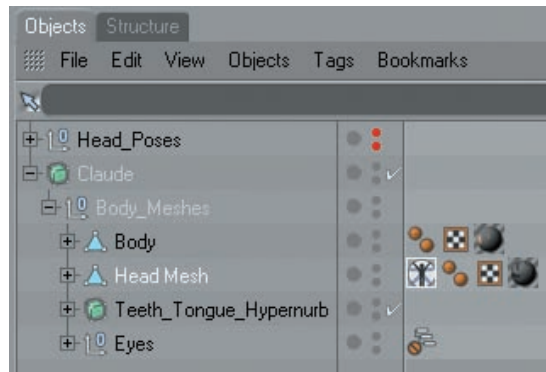
Open the file “QS_PM_Start.c4d”. It contains the character from the last chapter with new “Head Poses” object.

The new object contains copies of the figure’s head geometry including the head itself and the eyelids. “Head Standard Pose” is an exact copy of the original head. “Head Happy” is the same head with a smiling expression. In addition there is one head with raised ears and four heads, each with one closed eyelid.



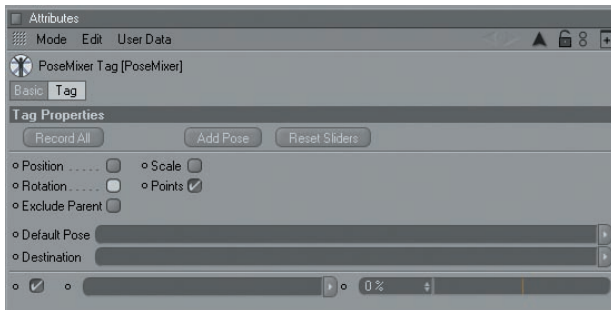
These new facial expressions were modeled based on the old head. It is very important that the copies have the same number of points as the original. You will see why we need these copies soon.

Find the original head in the hierarchy in the Object Manager. You can find it faster using the “search” tool (View/Show Search Bar” in the Object Manager or Ctrl+F with the Object Manager selected). Simply enter the name “Head Mesh” and the object will be displayed in the Object Manager. Right click on the object and select “Character Tags/PoseMixer”. A PoseMixer tag will appear next to the head object..



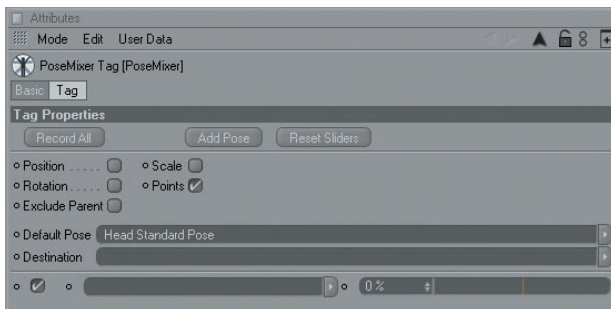
The PoseMixer works independently of hierarchies so it makes no difference which object we assign it to. It helps to stay organized, though, if you assign it to an object that it will morph. You will find the PoseMixer in the Attribute Manager.

Deactivate the options “Rotation” and “Scale”.



We have to let PoseMixer know what it has to morph. In this facial animation only the points of the geometry will be used. If we were to activate the option “Position” the entire head would move to where its copy is. Next, PoseMixer will need a neutral starting point, an undeformed state.

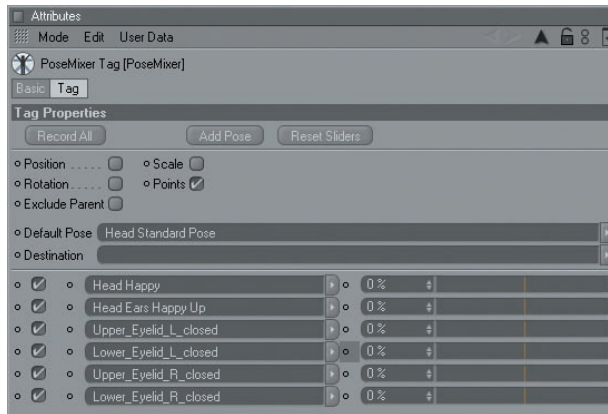
Drag the object “Head Standard Pose” from the Object Manager into the text field “Default Pose” in the PoseMixer’s Attribute Manager.



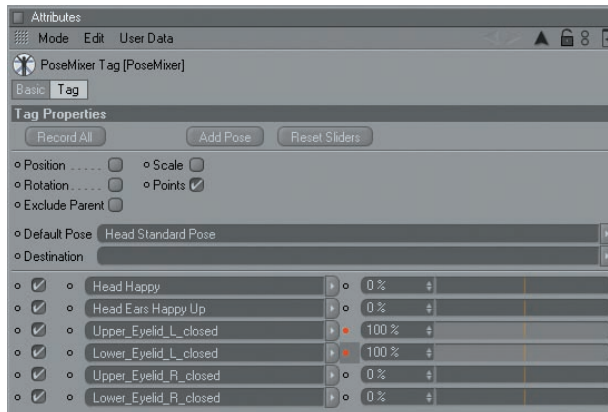
We can’t use the original copy of the head because it will be the one we will be deforming so we are using an exact copy of the head. The object to be morphed belongs in the “Destination” field but since we have assigned it a tag we don’t need to drag it into the Attribute Manager. The last field in the PoseMixer’s Attribute Manager is an empty field with a slider next to it. Naturally one of the modeled poses will be placed here.

Drag the object “Head Happy” into the field next to this slider. Now for the big moment: drag the slider and watch how the head changes its expression. The other heads will be placed into the PoseMixer as well, but first we have to make some room.

Click on “Add Pose”. Drag the “Head Ears Happy Up” into the new field. Repeat this process until all six poses are in the Attribute Manager.



What's nice is that the poses can be mixed in any order – try it. A relatively small number of base poses lets you create a wide variety of facial expressions. As when animating any other parameter, make sure that your scene is set to frame 0. Click on “Reset Sliders” in the PoseMixer Attribute Manager and then ctrl-click on the little black circle to the **right** of “Upper_Eyelid_L_closed” (belonging to the slider) and “Lower_Eyelid_L_closed” (**not the black circle on the left!**). The position has been keyed. Go to frame 5 and ctrl-click on the same circles again. Go to frame 3 and set both sliders to 100% and ctrl-click on both circles again.



You have just animated the wink of an eye. You can find the result in “QS_PM_Final.c4d”. By the way, PoseMixer is excellent for morphing entire bone hierarchies. You can set a variety of states for the hand bones – fingers stretched, fingers bent, – and mix them by activating the PoseMixer’s “Rotate” option. This saves the animator a lot of time since he or she doesn’t have to move each individual bone.

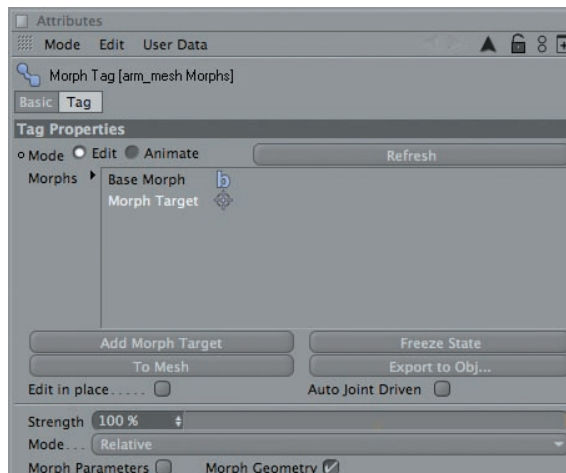
6. Quick Tutorial – Morph Tool

Now we'll turn our attention to the Morph tool. As previously mentioned, it is similar to the PoseMixer except that it makes it easier to create poses. There's no need to create copies of the original mesh when using the Morph tool. A single tag is all that's needed to store the poses (the Morph tag). Among other things, this means file sizes are much smaller than when using PoseMixer.

In this chapter, we'll take a closer look at the Morph tool and on the way explore the joint driven morph function. The basic principle when creating facial poses is simple: add a Morph tag to the mesh, create "morph targets" and move the points around to model the emotions in the morph targets. Simple. But there's more...

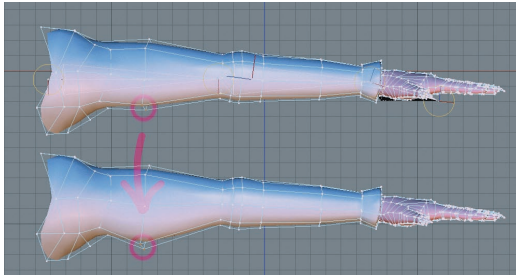
Open the file "QS_Morph_Start.c4d". This file contains two animations for the same arm. The only difference is that the left arm uses a joint driven morph to improve the deformation and simulate muscles. In this case, the morphing is linked to the rotation of the second joint. Rotating this joint by -90 degrees around its X-axis morphs the mesh to the morph target.

Once you've finished looking at the example, open the file "QS_Morph_Final.c4d". This contains the animated arm before it has been morph-optimized. We need to add a Morph tag before we can use the Morph function. Create the tag by right-clicking on the "arm_mesh" in the Object Manager and choosing Character Tags/Morph. Notice that the Morph tag already contains a base morph and morph target in the Attributes manager — they were assigned automatically.

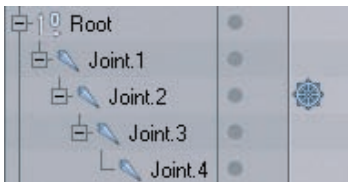


The base morph is our starting position and cannot be deleted. The morph target is the target pose you need to model. Unlike the base morph, you can freely create and delete morph targets.

Choose Cameras/Top in the editor to view the arm from above. Select the Morph tag if it isn't already selected and enable the "Auto Joint Driven" option in the Attribute Manager (more about this option later on). Click on the "Morph Target" in the Attribute Manager and select the Points tool. Select two points in the area of the biceps and move them down a little.



Next we need to select and rotate the joint that will be used to drive the morphing. In our case, this is joint number 2. Select the Model tool in the left toolbar and click on “Joint.2” in the Object Manager. Set the joint’s H rotation to -90 degrees in the Coordinates Manager and click the “Apply” button. Next, disable the “Auto Joint Driven” option. So what have we achieved by doing this? Earlier we enabled “Auto Joint Driven” in the Morph tag. When this option is active and you rotate a joint, a Driver tag is added to the joint.



Among other things, the Driver tag stores how many degrees the joint has been rotated and which axis the rotation is around. It passes this information on to the Morph tag, which in turn performs the morphing. Each time we rotate the second joint by -90 degrees around the H angle, the Morph tool will morph to the pose we created. If it wasn’t for the “Auto Joint Driver” option, we’d have to manually add a Driver tag to the second joint and adjust the tag’s settings. But in our case, all this took place automatically when we rotated the joint.

Select the Morph tag and in the Attribute Manager, set the “Mode” to “Animate”. This hides all the Edit mode functions and reduces it to a slider and a few functions. Set the slider to 0% and once again select our “Joint.2.” That’s it. Rotate the joint around its negative H angle and the muscle moves.

The possibilities here are virtually limitless. You can fine-tune deformations, prevent the geometry from shrinking or simulate muscles.

These are just a few examples of useful things you can do with the Morph tool besides creating facial poses. Morph offers many other useful functions. Suppose you’ve created most of the facial poses and then you realize you’ll need to remodel the face. This represents no problem for Morph! You can even add completely new details using the Knife tool without having to redo the existing poses.

7. Quick Tutorial – Muscles

Welcome to Muscles, the personal fitness training program for your CINEMA 4D characters. If your character has been hanging around idly because it hasn't been animated for so long, now it's time to give it some muscles. With MOCCA V3, you can take character animation to the next level with virtual muscles that flex beneath the skin.

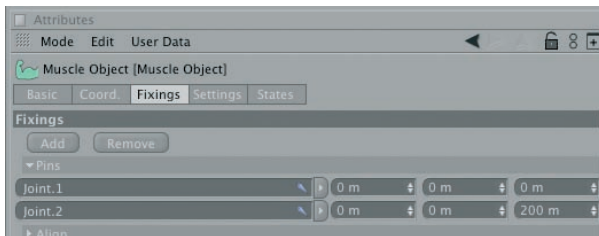
You need the following for a basic muscles setup: a Muscle object that is “pinned” between two joints, and a Muscle deformer that is a child of the polygon object the muscles should affect.

Let's start with a scene that shows you muscles in action. Open the file “QS_Muscles_Final.c4d”. The muscle deforms the geometry to simulate realistic muscle movement. For a further example of muscles at play, open the file “QS_Muscles_Final_Joints.c4d”.

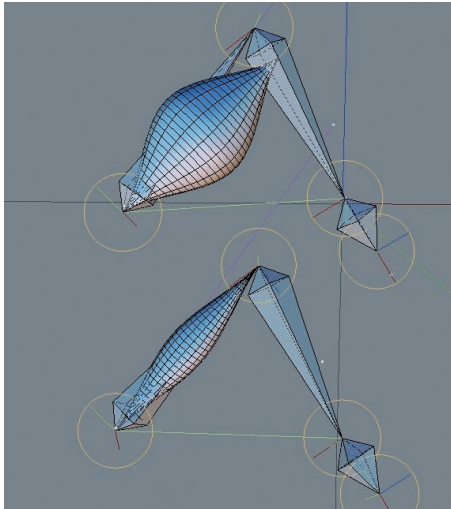
Once you have finished taking a look at the examples, open the file “QS_Muscles_Start.c4d”. This is our starting point. It's time to add some muscle.

Create a Muscle object (main menu/Character/Muscle Object). In the Attribute Manager, on the Fixings tab, you'll find two empty fields under “Pins”. Drag “Joint.1” from the Object Manager into the top field and drag “Joint.2” into the bottom field.

You can see the result immediately in the editor. We've told the muscle where it should start and end, but the default offset of 200 for the second pin's Z-axis makes the muscle jut out so that it's not between the two joints as we'd like it to be. Set the value from 200 down to 10.

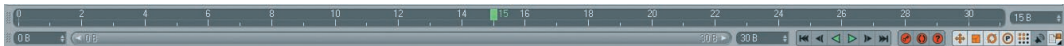


The muscle now lies exactly between the two points, but why the offset of 10? This parameter is extremely important for the muscles to work correctly. To see why it's necessary, open the file “QS_Muscles_Comparison.c4d”. In the bottom example, you can see that pin number 2 (the end of the muscle) is exactly on “Joint.2”. This prevents the muscle from being able to contract when the arm bends. In the top example, pin number 2 is away from “Joint.2” and this allows the muscle to contract and bulge as the arm bends. Select the “QS_Muscles_Start.c4d” file from the Window menu.



Switch to the Muscle object's States tab in the Attribute Manager. Open "Min" and "Max" by clicking their small gray triangles. The two curves shown control the shape of the muscle when stretched and contracted. But first we need to set the minimum and maximum length for the muscle.

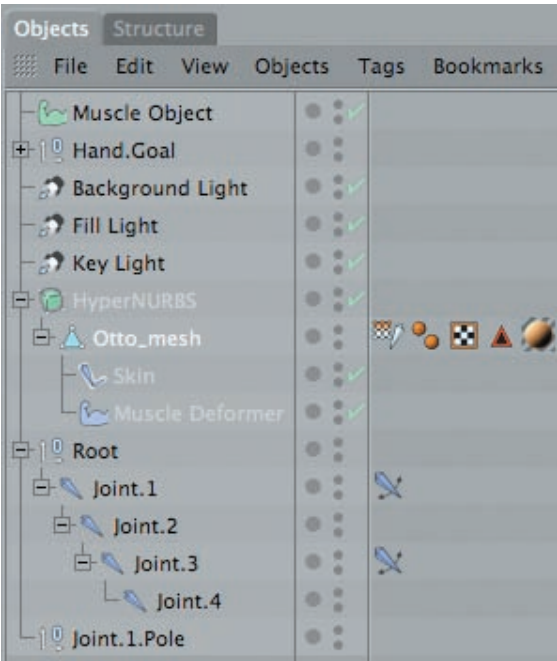
Make sure that the animation is at frame 0 (the scene already contains an animation). Click on "Set Max" in the Attribute Manager. This stretches the arm to automatically find out the longest length the muscle needs to be. This length is entered automatically into the Length field. Next, move to frame 15 of the animation.



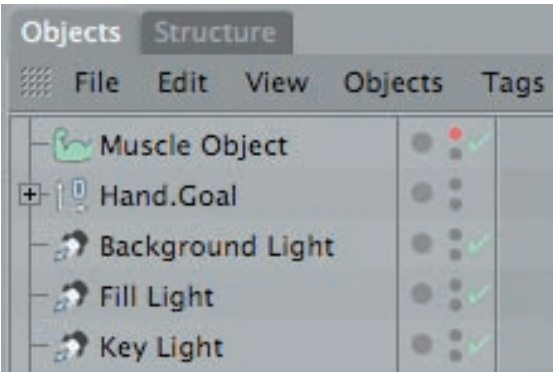
The arm is now bent. Click on "Set Min" in the Attribute Manager. Again, the muscle's current length is stored. The only parameter left to set now is "Strength". We will set the top "Strength" parameter to 130% because that's exactly how thick we want it to be when contracted.

It's the bottom "Strength" that needs to be adjusted. Set it to 20% so that it's reduced to 20% of its full thickness when the muscle is stretched. Click on the "Play" button in the animation toolbar. The muscle should bulge as it contracts. But what's the point of the muscle getting thicker when it's the skin that needs to bulge? After all, the muscle itself shouldn't be directly visible later on. This is where the Muscle deformer comes in.

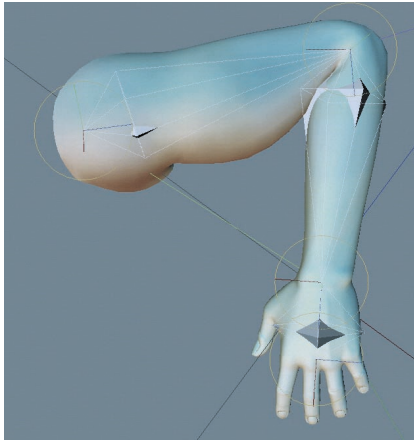
Stop the animation and choose Character/Muscle Deformer in the main menu. We need to make the deformer a child of the polygon object that is being deformed, which in this case is the object "human_arm_mesh". For the deformation to work correctly, the order in the Object Manager should be: polygon mesh / Skin deformer / Muscle deformer.



Select the Muscle deformer object. You'll find an empty field on its Object tab. Drag the "Muscle Object" from the Object Manager into this field. This tells the Muscle deformer which Muscle object it is responsible for. In the Object Manager, set the Muscle object to hidden (the top of the two small grey dots next to the word "Muscle Object" must be red) and play the animation once more.



Congratulations! You've just set up the arm to have a bulging biceps.



Experiment with the Muscle object's settings. As previously mentioned, you can change the shape of the muscle on the States tab using the curves. On the Settings tab, you can, among other things, move the muscle's offset without affecting the start and end pin.

As this tutorial demonstrates, adding muscles to your characters is a relatively simple task. Now there's nothing to stop your characters from showing off their muscles.

8. Quick Tutorial – VAMP

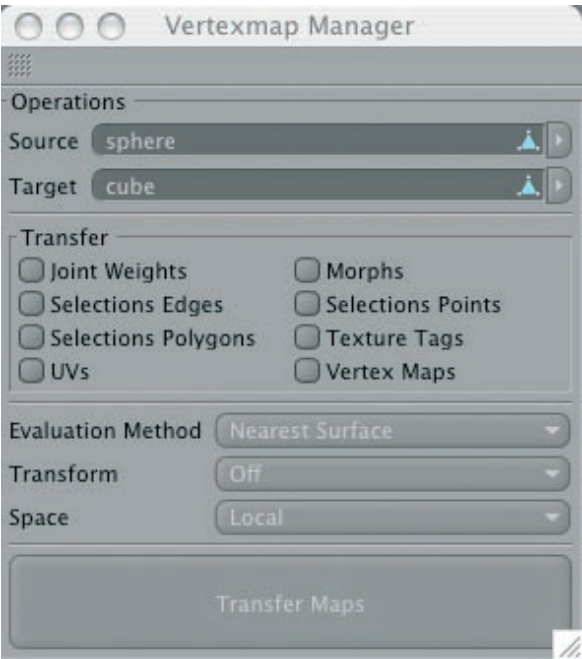
Vamp is a very powerful tool for transferring information from one object to another. It can transfer not only selection information, Texture tags and vertex maps but also morph data and UVs. A short example will help to explain why this tool is useful.

Open the file "QS_Vamp.c4d". In the editor, you'll see a sphere with a red cap, and in the Object Manager, there's a second object in the scene that has been set to invisible. This hidden object is a simple cube. We want to transfer the red texture and morph data from the sphere to the cube.

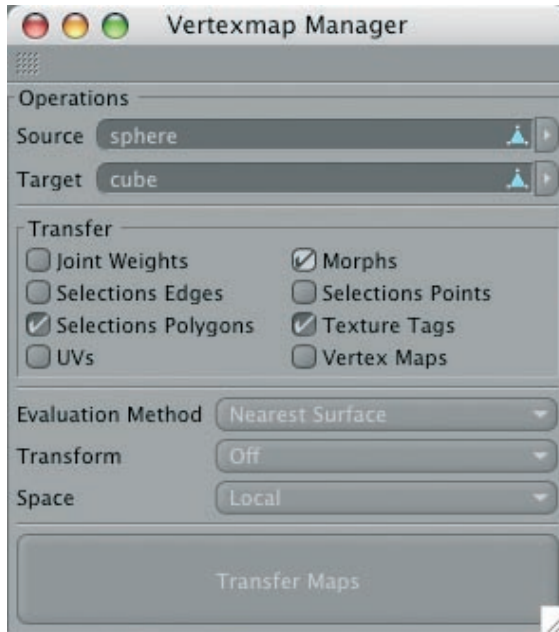
Select the sphere's Morph tab in the Object Manager. Set the "laugh" slider in the Attribute Manager to 100% and notice what happens in the editor.



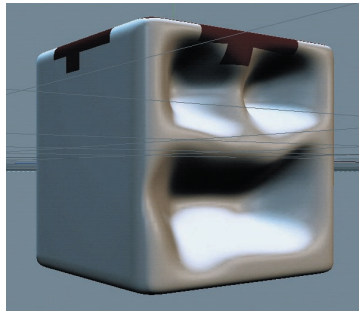
The ball is now smiling at us! Suppose we want to make the cube smile the same way using the sphere’s smile. Choose Character/Vamp in the main menu. A window opens containing several settings.



Drag the sphere object from the Object Manager into the Vamp tool’s “Source” field. Drag and drop the cube object into the “Target” field. We’ve now told Vamp which object the information should be transferred from and which object it should be transferred to. Enable the options for “Selections Polygons” and “Texture Tags”. Click on the “Transfer Maps” button.

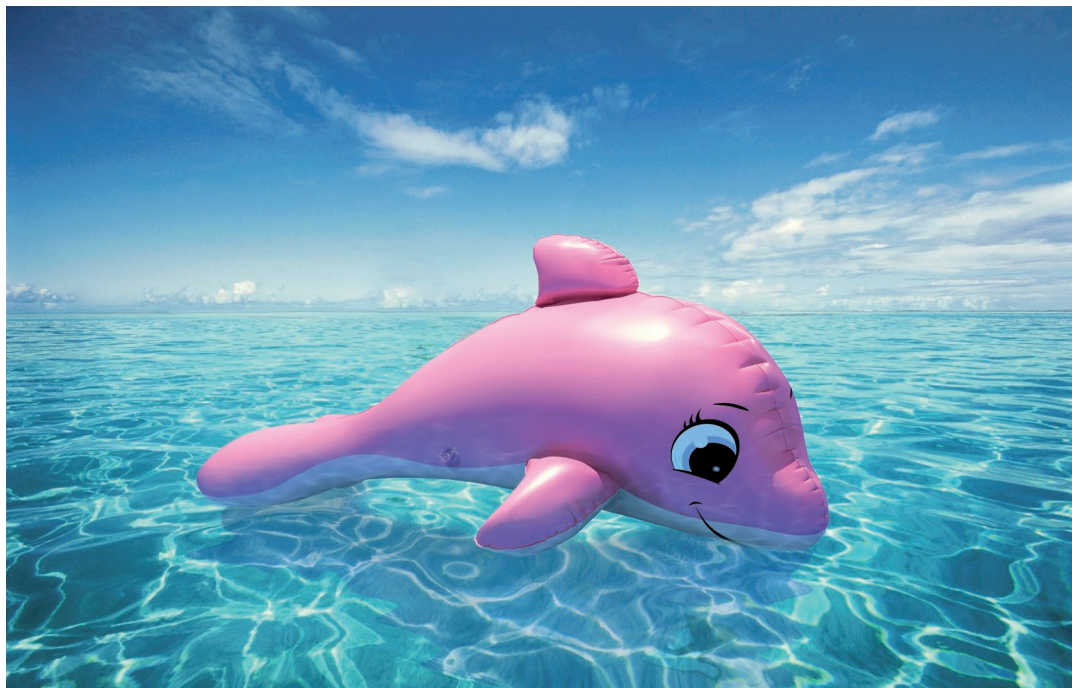


If you now look to the right of our cube, you'll notice it has a Selection tag, a material and a Morph tag. Make the cube visible by clicking on the small red dot next to the word "Cube" in the Object Manager. Click on the top dot for the sphere until the dot turns red.



As you can see, the cube now has a red cap which has been transferred very accurately, even though the cube has a different number of polygons to the sphere. If you now select the cube's Morph tag, switch to the "Animate" function and change the Strength slider in the Attribute Manager to 100%, you'll see that even the morph data has been transferred. The cube is now smiling at us just like the sphere was!

Vamp makes it quick and easy to perform these kinds of transfer. The possibilities are limitless. Further experimentation with this tool is highly recommended.



© Platinum, www.platinum.com.br

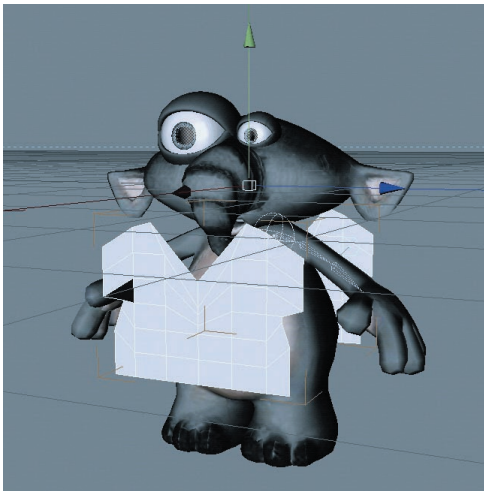


9. Quick Tutorial – Dressing

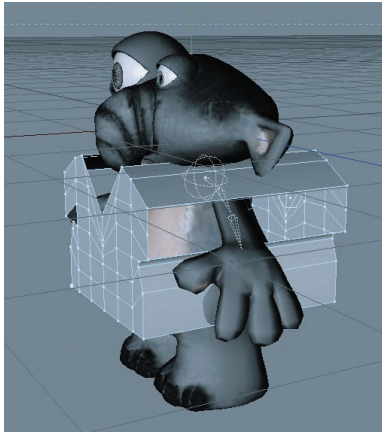
Clothilde is MOCCA's cloth simulation tool. You can use this tool to let a flag flap in the breeze or to give your character a snappy t-shirt. This is exactly what we want to put on Claude.

What's nice about Clothilde is that it's not necessary to go through the trouble of modeling a t-shirt. All you have to do is create the front and the back of the shirt. Clothilde will make it fit automatically.

Open the file "QS_Cloth_Start.c4d". This file already contains both t-shirt halves as a single object.



If you don't like the shirt feel free to replace it with your own creation. As you can see it was created with very few polygons. Select the shirt. Switch to point mode and select the Bridge tool (Structure/Bridge). Connect the front and back of the shirt (of course only where the shirt would really be connected – the openings for the sleeves, head and body should not be connected). The Bridge tool works as follows: Simply click on a point on the front (or rear) shirt surface and drag to the point lying directly across from that point on the other shirt half. Repeat this process with the next (neighboring) point. As you can see, this creates a surface between the two shirt halves. Repeat this process until both halves are connected completely.



The shirt needs to be subdivided a little more so you can deform it better later. Switch to “Use Polygon Tool” mode and select the polygons on the front and back side of the shirt. Select “Functions / Subdivide” and set the value to 2.

Make sure you don’t subdivide the polygons on the side (the ones created by the bridge tool). OK, it doesn’t really look like a t-shirt yet, it looks more like a box with sleeves. But don’t worry, we’re going to change that right now. Clothilde is very easy to use. Most functions are contained in one tag, the “Cloth Tag”. It will be assigned to the object, which will then be turned to cloth!

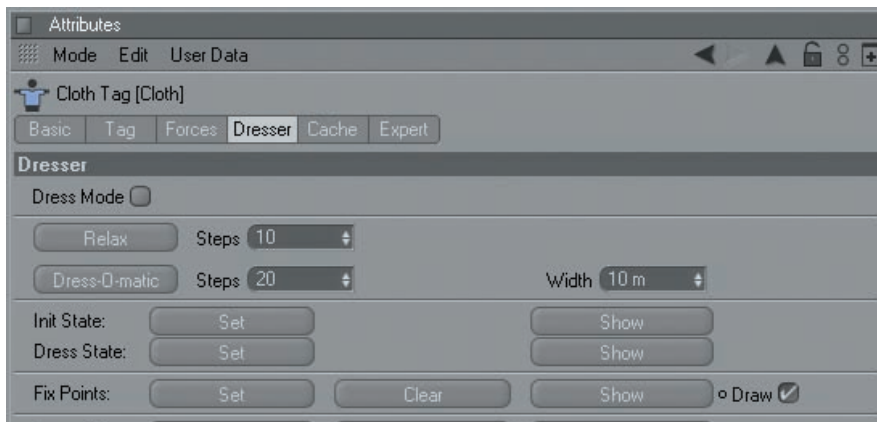
Right-click on the shirt and select “Clothilde Tags / Cloth Tag”. The Clothilde properties will appear in the Attribute Manager.



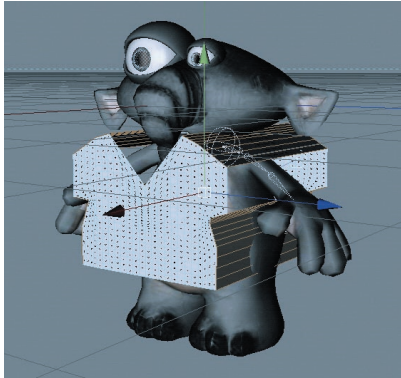
If the cloth object is to collide with another object, as is the case with the t-shirt and the body, the other object must be assigned a collision tag. Select the character’s body (polygon object “Body”) in the Object Manager with the right mouse button. Select “Clothilde Tags / Collider Tag”.



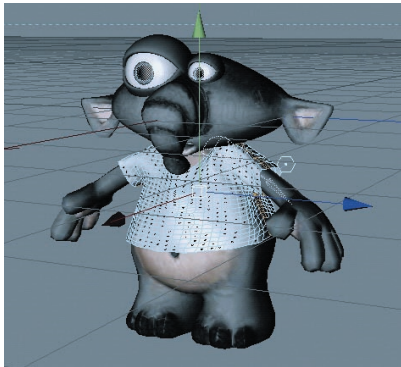
Now the t-shirt knows that it should not pass through the character. It's about time that we gave the shirt its proper shape. Select the cloth tag and switch to the "Dresser" menu.



In the dresser menu you will find everything you need to make clothes fit. The other menus deal with the cloth's behavior. Select "Set" next to "Dress State". This is like a security measure. We can recall the shirt's initial state in case we don't like the position of the shirt or if we want to add a breast pocket, for example. Switch to "Use Polygon Tool" mode and select the polygons on the sides (the ones created by the bridge tool). Alternatively you can select "Selection / Invert Selection" in the C4D main menu - since the front and back side of the shirt should still have been selected, this can save you time.



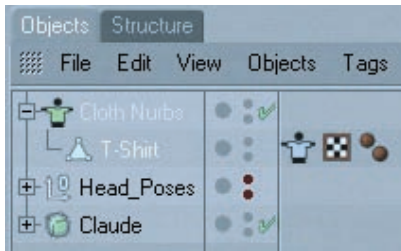
These polygons will serve as the t-shirt's seam. Clothilde will do this for us as well. Select "Seam Polys" in the "Dresser" menu and set it to "Set". The seam is still a little too wide. Click on "Dress-O-Matic" and look what happens with the T-shirt.



The seam will be "pulled together" in accordance to the "Width" value. It doesn't necessarily match the shape of the character's body, though. The "Steps" value determines how exact this fit will be. After setting the "Init. State", click on "Relax".

Now other forces such, as gravity, will have an effect on the T-shirt which makes it sag on the character's shoulders.

Place a "Cloth Nurbs" into the scene (Character/Cloth NURBS). Make the t-shirt a child of this object.



Cloth nurbs acts similar to a HyperNURBS object: it smooths the geometry which was subordinated to it but with a slightly different algorithm which works better for cloth objects. Additionally, a thickness can be specified for cloth nurbs fabric objects.

Set the “Thickness” to 1 in the Attribute Manager and the subdivision to “0”. Create a HyperNURBS object and make the “CNURBS” object to a Child object of this HyperNURBS object. Now the “CNURBS” object will concentrate on the thickness of the cloth and the HyperNURBS object will take care of the t-shirt’s edges.

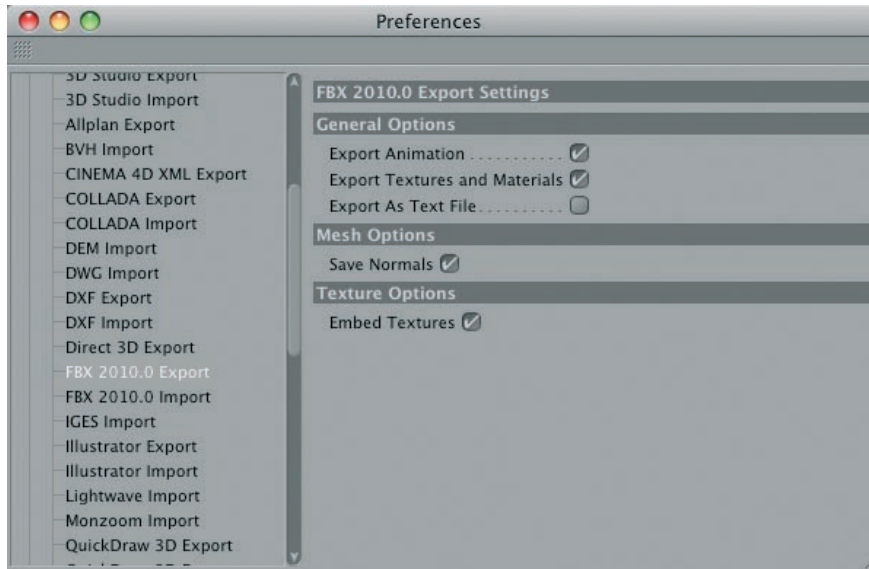
The character should be clothed now. Of course there is still some fine-tuning necessary to make the t-shirt behave properly in an animation. As you can see, though, the first steps weren’t that difficult.

10. Quick Tutorial – FBX Import / Export

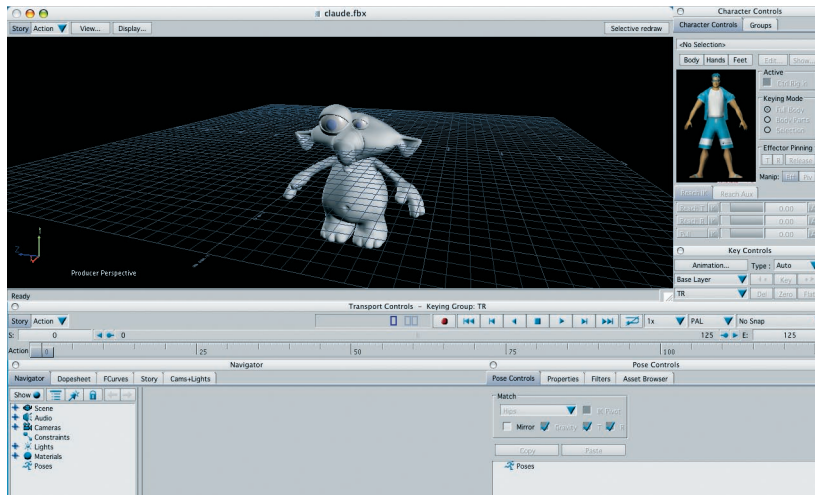
The FBX file format is used by most 3D programs to import and export files. It can be used to exchange complete scenes and animation files. It is a standard format for motion capture. Motion capture is the recording of human (or animal) motion using special sensor systems. The files can be assigned to just about any virtual character using the FBX file format. Maybe you want your character to behave just like you? If so, pay close attention to the following information about FBX.

Let’s assume you have a character with a complete skeleton and has the correct strengths but you don’t want to animate it manually. You want to use an animation that has already been created, maybe a walking animation. “Motionbuilder” is a software that was conceived for this purpose. This program lets you assign movement data to a character. Since CINEMA 4D can’t read this file format the character must be exported as an FBX file.

Select “Edit / Preferences / Import-Export / FBX Export”. Here you can determine what it is you want to export. In most cases the default settings can be used.



If all the settings have been made the export can begin. Select “File / Export / FBX”. Save the file as a FBX file. Now a program like Motionbuilder can open the CINEMA 4D character.



Motionbuilder assigns motion capture data to the character. These are then “plotted”, which means the animation is converted into movement and is assigned directly to bones. The final animation will be saved as

an FBX file which CINEMA 4D can easily read.

Select "File / Merge. Load the FBX file.

It's not necessary to open the complete file. The FBX import is smart enough to know that it only has to load the animation. Don't be surprised when you look at the timeline: A key has been set for every object on every frame.

Of course you have to take into account a few more things when exchanging files with Motionbuilder. It makes life (and working with a character in Motionbuilder) easier if you use a naming convention when naming your bones. You can get more information about this from MAXON or the makers of Motionbuilder.



© S. Scatola, www.boxy.co.uk



© www.bediff.com



Welcome to HAIR (Optional module)

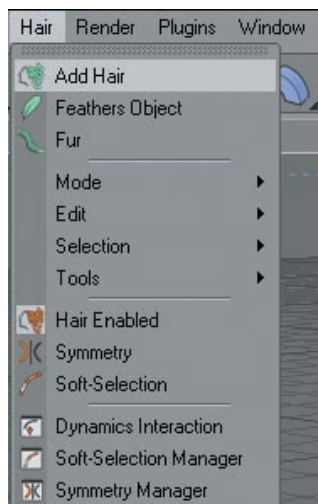
This tutorial will show you how you can enter the world of HAIR and easily create hair, fur and feathers.

1. Introduction

HAIR is a powerful tool for creating various types of hair and fur. Even feathers, animated grass, anemone and much more can be created using HAIR.

Although achieving your first results in HAIR is relatively easy, HAIR is an immensely diverse and comprehensive module that will satisfy any beautician.

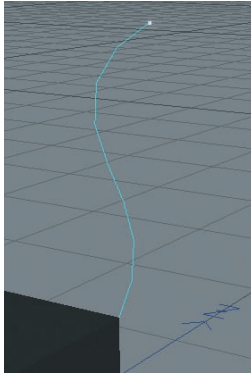
HAIR can be accessed from the main menu at the top of your standard interface.



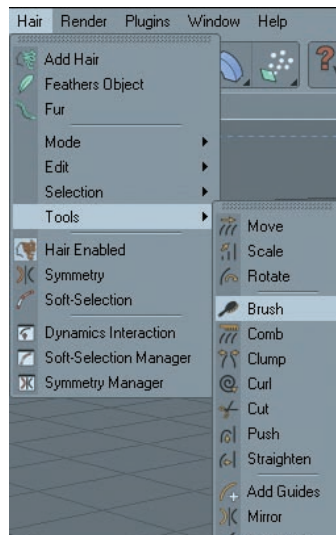
HAIR's only limitation is your fantasy, whether it's creating fur for a rodent, feathers for a ruffled chicken, the perfect English lawn or the newest hairstyles. And, HAIR IS fast – what else have you come to expect from CINEMA 4D? HAIR renders immense amounts of hair with unmatched speed. The variations that HAIR offers are so great that any hairstyle can be created, from smooth and straight to curly or just about anything you can imagine.

2. General Information / Interface

HAIR works with so-called guides that serve as placeholders for the rendered hair.



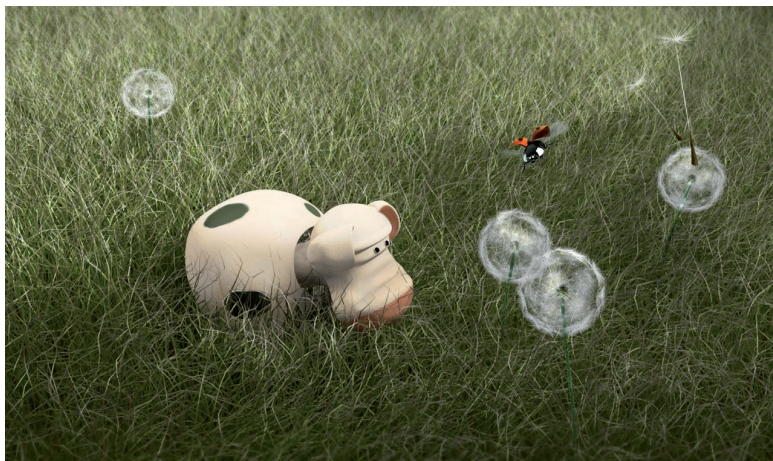
The number of guides displayed in the editor view is far less than the actual number of hairs that will be rendered. The number of guides displayed can also be increased. The “missing” hairs are interpolated between the guides when rendered. Naturally, you will require some standard grooming tools to bring your hair into shape. Among the tools HAIR offers are Brush, Comb, Scissors and more.



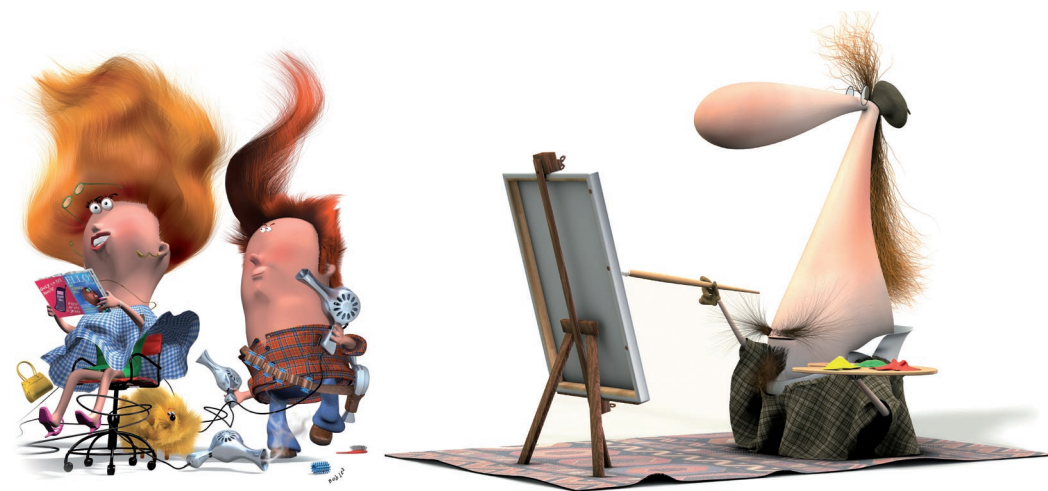
HAIR’s own IK makes sure hair moves realistically. Even some of the CINEMA 4D particle modifiers, e.g. wind, can be combined with HAIR and the HAIR dynamics ensure hair behaves naturally. And if you want to transplant hair, that’s no problem, either. HAIR lets you easily transplant hair roots.

Before we get started with the HAIR tutorial, though, let's take a look at what can be created with this module:

3. Sample Images



© Patrick Eischen, www.patrickeischen.com



© www.boblea.com

© Bill Ledger, Toybox Animation

Now let's get started with the HAIR tutorial...

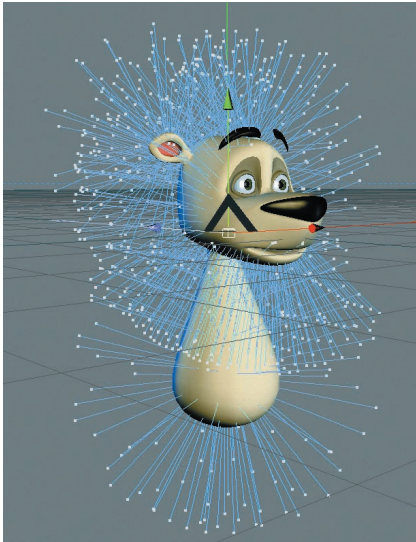
4. Quick Tutorial – Fur

Our volunteer, “Hairbert”, is only a few mouse clicks away from becoming his warm winter fur. He may look a little pitiful without his fur but his simple construction will make our work that much easier.



First, open the file “QS_Hair.c4d”.

HAIR can be applied to either an entire object or a polygon selection only. Since we want to give Hairbert’s face a different fur than the rest of his splendiferous body we will only select those polygons to which our first fur type (long fur) should be applied. Go to the Object Manager and select the Bear_mesh object. Make sure the Use Polygon Tool is active in the Tool Palette at the left of your interface. In our example the corresponding polygons should already be selected. All polygons onto which long fur should be placed on Hairbert’s body will be highlighted by the orange selection. Select Hair / Add Hair in the CINEMA 4D main menu. Subsequently, the guides we mentioned at the start of this tutorial can be seen protruding from Hairbert’s body.

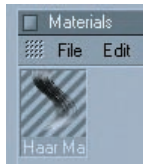


The guides all protrude perpendicularly from each surface and have a default length of 100. Go ahead and Render the scene to see what poor Hairbert looks like (Render / Render View or the far-left render icon).

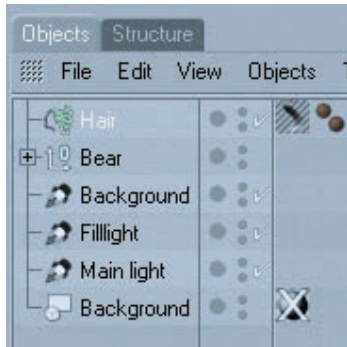


If Hairbert were a porcupine or had just come out of the spin cycle at 90° we could finish this tutorial at this point. Since Hairbert deserves better we will continue and use the settings described below to give this guy his cuddly winter fur.

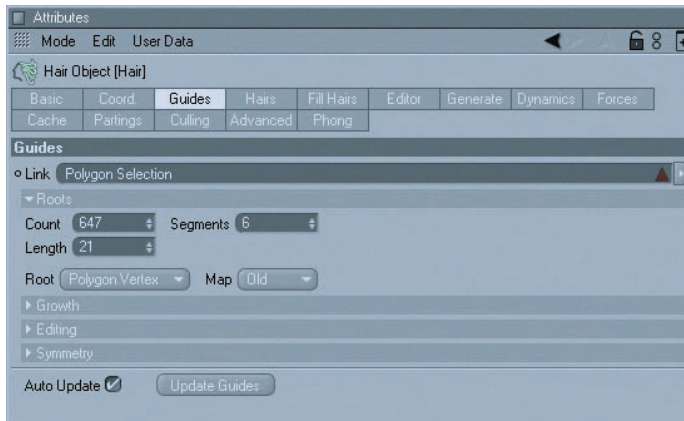
When hair is added, a corresponding material will be created automatically in the Materials Manager.



The HAIR object is located at the top of the object list in the Object Manager.



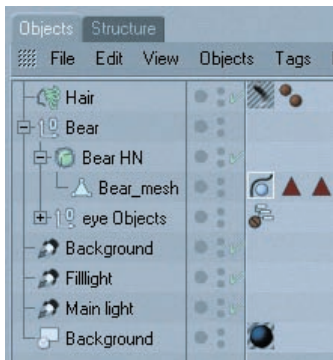
Before we start editing the HAIR material and the HAIR object, we will shorten the guides a little (which will also shorten the rendered hair) and give the hair a little style by applying gravity. Select the HAIR object in the Object Manager. Go to the Attribute Manager and activate the Guides tab. This tab contains the Length and Segments settings.



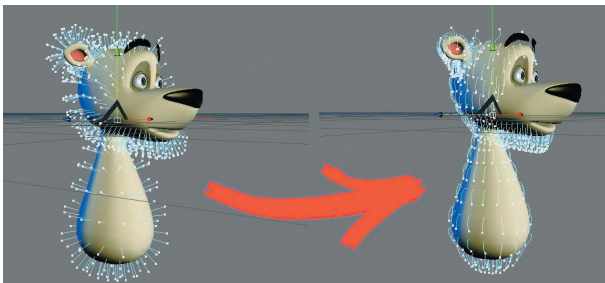
Set Length to 21 and Segments to 6. The Segments setting defines the number of individual segments each guide is made up of (this only influences how guides are displayed in the editor, not the rendered hair). If Segments were to be set to 1 each guide would consist of only a single segment and gravity could not bend the guide. Our guides have 6 segments, which means they can be bent at 5 points along their length (see the Tips & Tricks chapter for further details concerning guide segments).

Now switch to the Forces tab and activate the Surface to Hair setting. This will prevent the hair tips from penetrating any surfaces when gravity is applied (collision detection). If we were to apply gravity at this point, though, the hair tips would still penetrate the polygon mesh since we have not yet told the polygon mesh that it should interact with the hair.

To do so, click on the Bear_mesh object in the Object Manager and assign to it a Collision tag (right click: Hair Tags / Hair Collider).



If you play the animation now you will see how the hair is pulled down by the gravity.



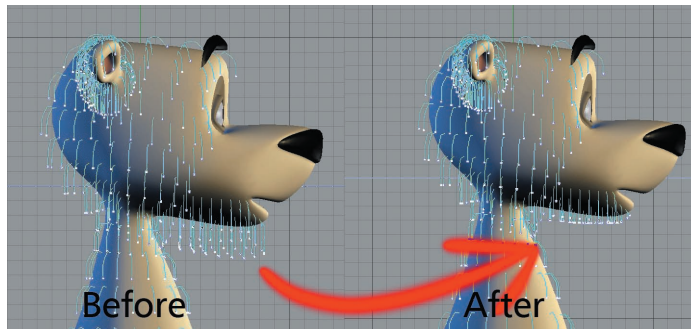
Stop the animation before it reaches frame 100 to prevent the calculation from starting anew. Alternately you can start the Dynamics calculation in the HAIR object by pressing the Relax button in the Dynamics tab's Animation menu. Click this button after the animation has been set to frame 100).

In order not to lose this state we have to tell HAIR that this should be our new initial state. Simply setting the animation back to frame 0 will set the calculation back to its original state and Hairbert would again look like a porcupine in shock.

To prevent this from happening select Hair / Edit / Set As Dynamics in the main menu. Now we can cut and style Hairbert's fur.

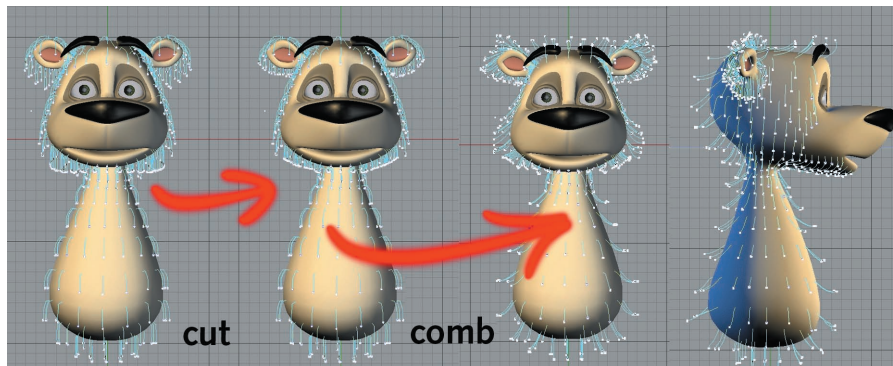
Switch to the left view in the editor window and activate the Cut tool (Hair / Tools / Cut). Next, deactivate Only Visible in the Attribute Manager to make sure hair that is not visible will be cut as well. Now set Form to Circle and let's start cutting.

To cut Hairbert's hair simply click and drag over the blue tips of his hair. Cut his hair as shown in the screenshot below:



Switch to the front view in the editor window so we can trim and style the hair on Hairbert's chin a little. Activate the Brush tool (Hair / Tools / Brush). Activate the Collisions setting (one of the Brush tool's most important settings (Options tab)) and set Coll. Radius to 1. Doing so will prevent the hair from penetrating Hairbert's skin when brushed. Leave the "Deactivate Visible Only" option disabled here as well.

Switching views as needed in the editor view can be very helpful when brushing or styling hair. To brush hair, simply click and drag over the blue hair tips. The guides possess their own HAIR IK which ensures hair-like behavior when the hair is styled. To create our fur we will need to pull the guides out a little and curl them up slightly at the tip. Take a look at the following screenshots to get an idea of what we mean.

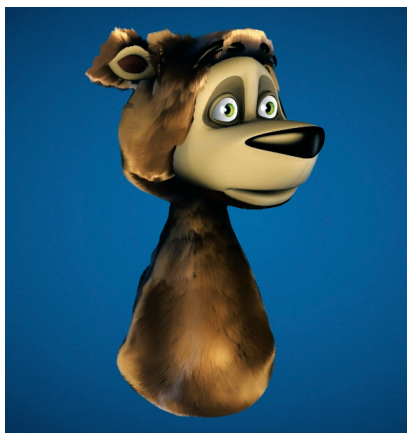


After you've had your fun with Hairbert's hair and it has the look you want switch to the perspective view in the editor window and position Hairbert so you can take a good look at him. Render the view and see what Hairbert looks like (main menu: Render / Render View).

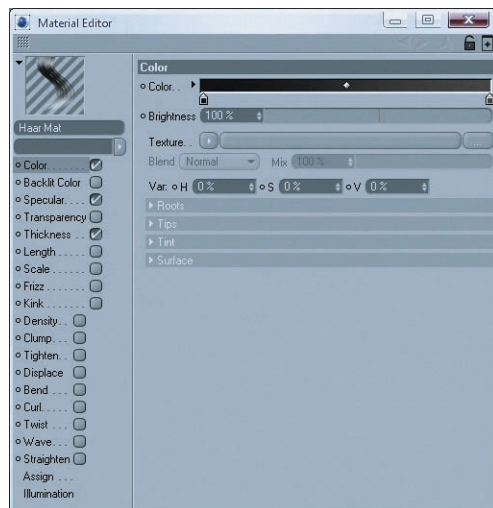


It seems that Hairbert is suffering from a minor case of hair loss. This is not due to stress but the fact that we still need to make some changes to his hair settings. We'll do that now.

First, we will turn our attention to the number of hairs Hairbert has. Select the HAIR object in the Object Manager and set Count to 100,000 in the Hair tab menu in the Attribute Manager. Activate Fill Hairs in the Fill Hairs tab and set Count to 80,000. Render the scene again. It should look similar to the image below. Poor Hairbert has been transformed from a porcupine to a bad Prince Valiant look-alike. We'll have to make more changes.



The properties of Herbert's hair still have to be edited extensively so let's start by double-clicking the HAIR material in the Materials Manager. This will open the material's dialog window with all its options and channels.



Follow these steps:

- Specular: Set Primary Strength to 34% and Secondary Strength to 50%.
- Thickness channel: Set Root to 0.5m and Tip to 0.03m. This will thin Hairbert's hair to make it look more realistic.
- Scale: Activating this channel will vary the size of the hairs, which adds additional realism. Set Scale to 100%, Variation to 80% and Amount to 40%.

Render the scene once again. The result should look like this:

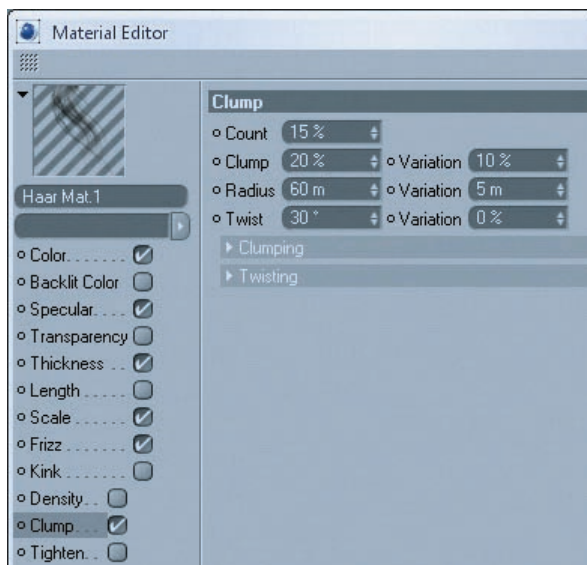


As you can see, Hairbert's bad hair day is slowly but surely coming to an end. His fur is starting to look much more realistic. We'll remove some of its smoothness and shine by adding a little frizz. Select the Frizz channel, located below Scale, and set Frizz to 15%.



Hairbert's looking better and better. Soon he can let himself be seen in public again. What we want to do now is get rid of that "just blow-dried" look and clump the hair a little. We will do this using the Clump channel.

Select the Clump channel and change the settings to match those in the screenshot below:



Render the scene again. The result should look like this:



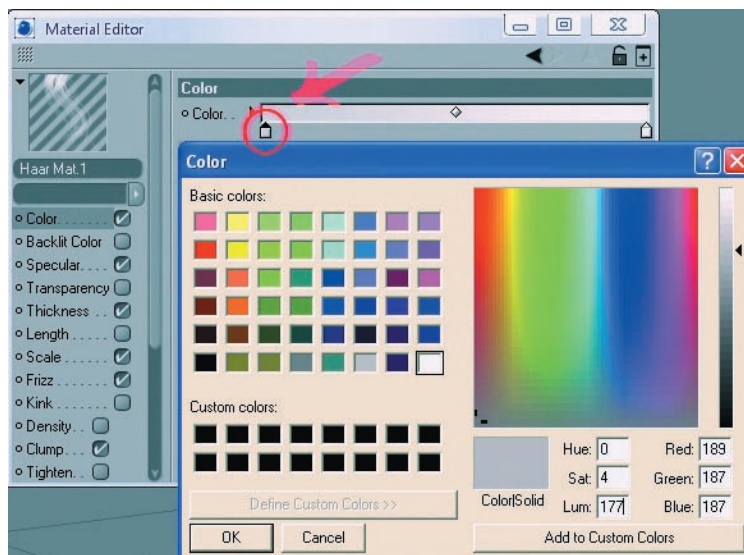
So what's left to do? Right, Hairbert's face still needs some hair. As mentioned at the beginning of this tutorial, we will apply a different type of fur to Hairbert's face. This time it will be even easier to apply! Again, we have pre-selected the polygons for you to which this fur will be applied.

First, make sure the Use Polygon Tool is active and select the Bear_mesh object in the Object Manager. The polygons to which we just applied Hairbert's long fur will be highlighted in red. First, click on the "Live Selection" tool in the top Icon Palette then click anywhere next to the figure in the editor window. All polygons will be deselected. Select the third triangle (from the left) with the red border next to Bear_mesh in the Object Manager. In the Attribute Manager you will see "Hair Short" in the Name text field. Click on Select Polygons below this text field. If you followed all steps correctly the polygons on Hairbert's face and ears should be highlighted in orange.

Since we just covered the following steps in the creation of Hairbert's long fur we will simply list them for you here. Give your new HAIR object (e.g. "Short Fur") a different name to avoid confusing it with the long fur.

1. Add Hair (main menu: Hair / Add Hair)
2. Set guide Length to 5, Segments to 4 (Attribute Manager)
3. Activate Fill Hairs.
4. Set Count for hairs and fill hairs each to 250,000.
5. Reduce hair (not guide) Segments to 4.
6. Open the HAIR material (double-click the material in the Materials Manager).
7. Set Root to 0.3 and Tip to 0.01 in the Thickness channel.
8. Activate the Straighten channel and set Strength to 70%.

Since Hairbert was born in Alaska we'll have to change his look a little. Double-click one of the HAIR materials in the Materials Manager and select the Color channel. Double-click the left-most color marker.



Change the color to a very light gray and click on OK. Now select the right-most color marker and change its color to white. Do the same for the remaining HAIR material and render the scene.

You may have to tweak the colors a little but your result should basically look like this:



Congratulations! You have just completed your first HAIR project. It's that simple! Now take what you have learned here and experiment with adding hair to other objects, transitioning from long hair to short hair, creating different styles, even creating grass and more. The only limitation HAIR has is your own imagination.

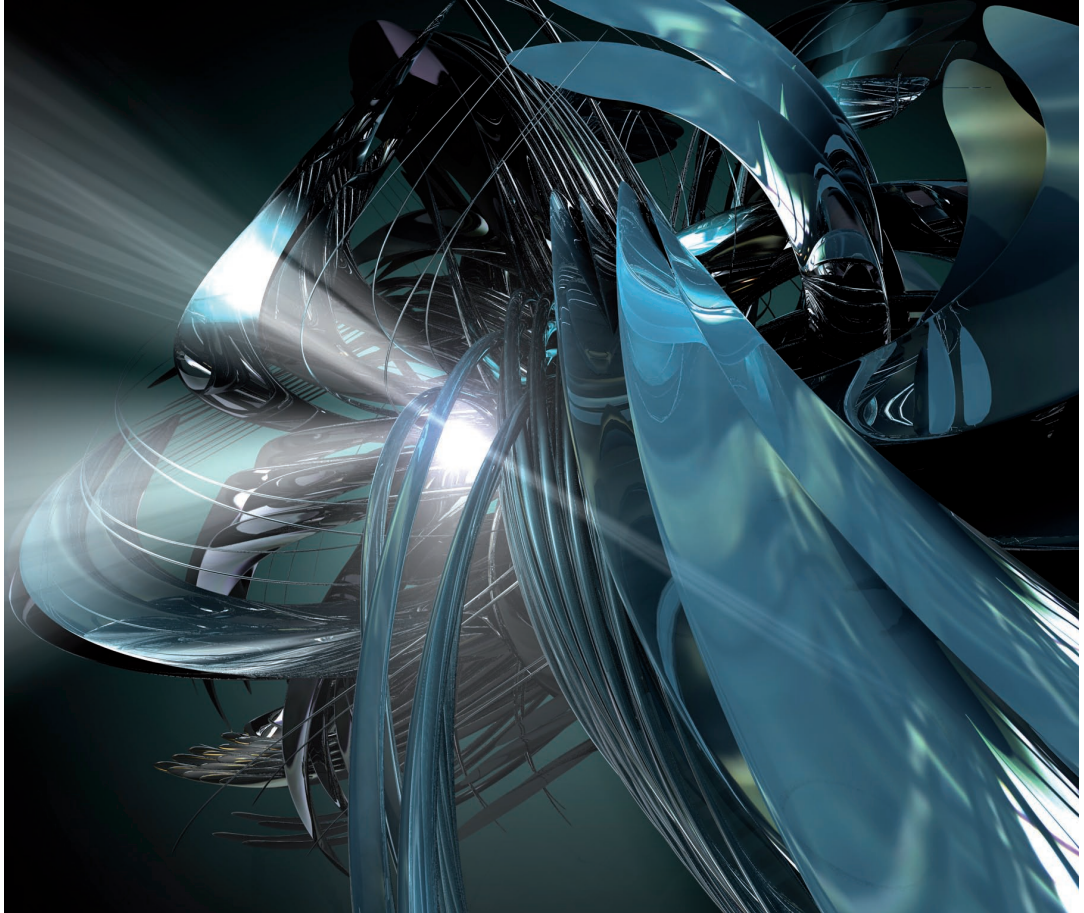


5. Tips & Tricks

- You can save a lot of time by optimizing your HAIR settings. Make sure your object really needs those 500,000 individual hairs – maybe half as many will suffice without sacrificing realism.
- A greater number of individual hairs are required when creating short hair (as was the case with Hairbert's fur) to prevent surfaces from showing through. Short hair, though, requires fewer segments because it is stiffer by nature. Increasing the segment count for short hair is therefore unnecessary.
- The method of creating hair used in this tutorial is by far not the only method for creating good-looking hair. You can, for example, experiment with the various HAIR modes when brushing hair (main menu: Hair / Mode / Points). Since guides behave differently when working in Point mode, as opposed to Tip mode, entirely new possibilities for styling hair are made available.
- Experiment with the guides' Segment setting. If you set the number of guide segments to 3 and the number of hair segments to 20 the hair would be very bouncy and protrude a greater distance from the surface. The rendered hairs would look round and have no corners. The hair will look like it hasn't been shampooed for several days.

Welcome To MoGraph (Optional module)

This MoGraph Quickstart tutorial will give you a detailed insight into the limitless possibilities MoGraph has to offer.



© Simone Bertolotti, www.imonfox.it

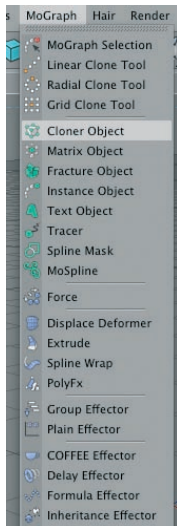
1. Introduction

The MoGraph module is designed to clone just about any geometric primitive or object and offers numerous Effectors with which these objects can be controlled. For example: Large objects (which are generally made up of several smaller ones) can be exploded, deformed and morphed into another object; many small objects can be controlled by a target object to which they have been assigned; a surface made up of various objects can be deformed using a shader; spheres can suddenly appear and come together to form a word. The possibilities are endless!

MoGraph offers creative minds a virtual universe of endless possibilities. Once you complete this tutorial we urge you to let your creative juices flow – experiment to your heart's desire and see what MoGraph can do!

2. General Information / Interface

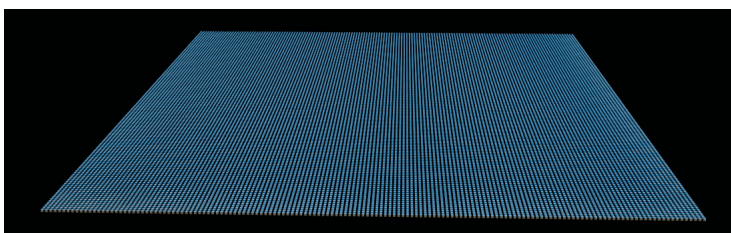
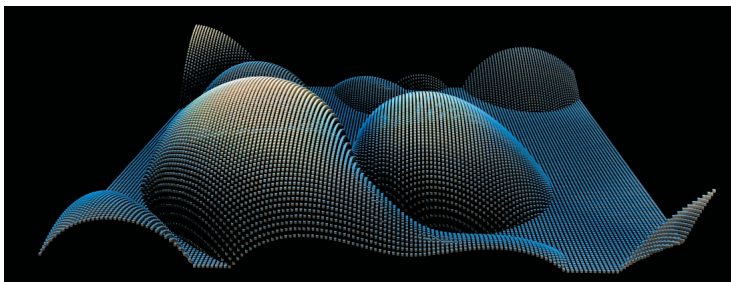
Once MoGraph has been installed it can be accessed in the CINEMA 4D main menu.



In most cases, a Cloner Object will be required in order to create a MoGraph scene. The Cloner Object contains all functions required for the creation and depiction of clones. The Cloner Object is then augmented by the various MoGraph Effectors. Below is a sampling of Effectors and how they work:

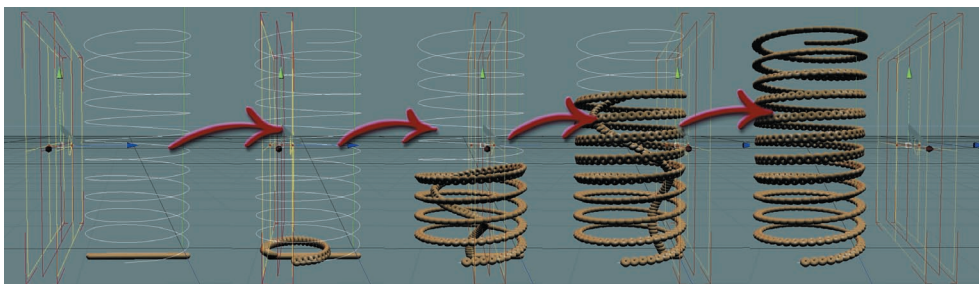
- **Delay Effector:** Use the Delay Effector to delay any action or keyed animation of a cloned object (see chapter 5, Tips & Tricks).
- **Shader Effector:** The Shader Effector analyses textures or shaders applied to clones and uses their height and depth information to deform the clones' surfaces. For example, a noise shader, including its grayscale values, can be loaded into a Shader Effector and subsequently be animated. This animation will then affect the clone onto which it is applied.

The following two screenshots show a clone with and without a Shader Effector with an animated noise shader applied.



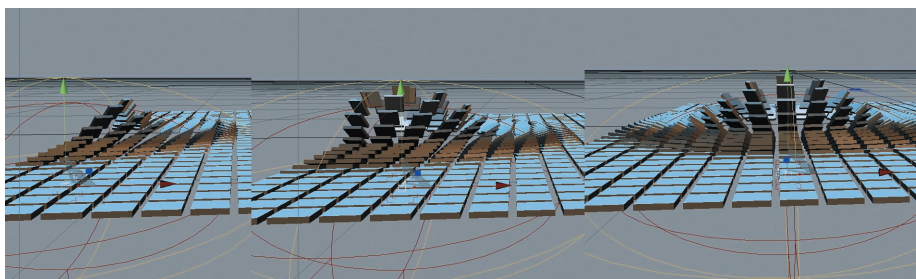
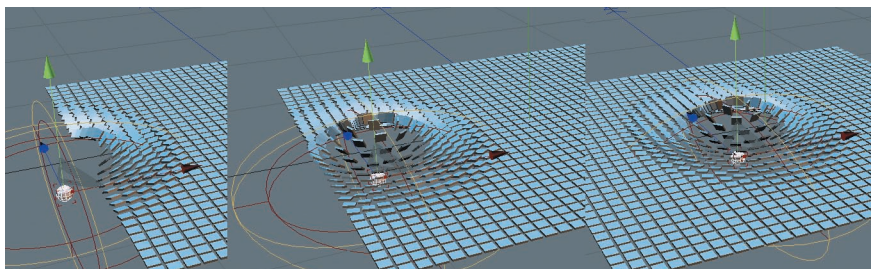
- Spline Effector

You can use the Spline Effector to link spline-based shapes or objects to the Cloner Object. Clones can be aligned to create text or morphed into text or other shapes. The image below shows how a Spline Effector, with Falloff set to Linear, (moved in the direction of its own Z-axis) was used to morph a row of “donut” clones into a helix shape.



- Target Effector

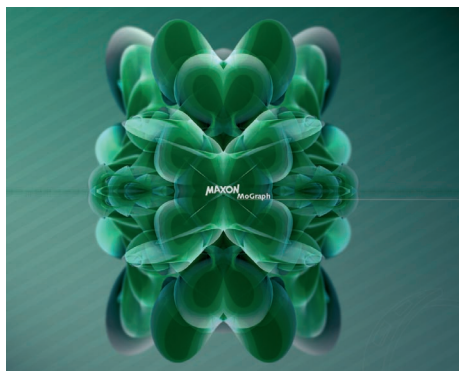
The Target Effector lets clones be aligned to a target object. The clones will follow the movement of the target object accordingly. In the image below, a ball was used as the target object, to which a Target Effector, with its Repel setting activated, was assigned.



This is also the scene we will use for this tutorial.

Before we get started, sit back and take a look at what MoGraph can do. Maybe these images can say more about what MoGraph can do than any amount of words...

3. Sample Images



Above images © MAXON Computer GmbH (David Drayton)

4. Quickstart Tutorial

In this tutorial we will show you how to achieve fantastic results with just a few clicks of the mouse. MoGraph has been designed to quickly master tasks that, until now, took hours to complete, if it could be completed at all.

For example, the Target Effector: How would you animate 1000 clones that follow a target object without the use of MoGraph? Well, we won't waste any time trying to answer that question. Instead, we will show you how easily it can be done using MoGraph.

We will require the following items for our scene:

- Cloner Object
- Target Object (a simple parametric object)
- Target Effector
- Primitive (the object to be cloned)

All we need for this tutorial is these four simple items – MoGraph will do the rest. All you have to do is be creative (as you surely always are...).

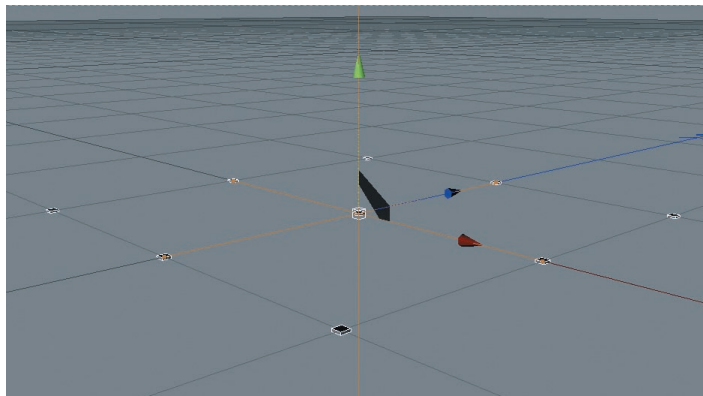
Create a Cloner Object (main menu: MoGraph / Cloner Object) and a cube (main menu: Object / Primitives / Cube).



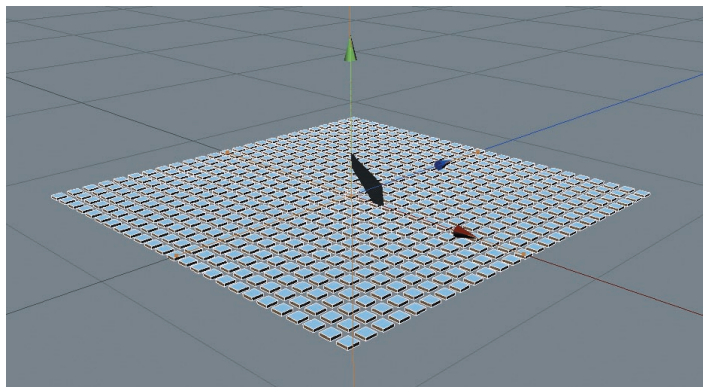
The cube will serve as the object to be cloned and will be made a child of the MoGraph Cloner Object in the Object Manager. Before this happens, though, the cube has to be resized. Set the size of the cube in the Attribute Manager to the right to: X=4; Y=1; Z=4. Once the cube has been rescaled you may have to zoom in to the cube a little to get a better view of your scene. In the Object Manager drag the Cube object onto the MoGraph Cloner Object. This will make the cube a child of the Cloner Object and simultaneously tell the Cloner Object to affect the cube.

If you didn't zoom in too far you will see that two additional cubes have been created (you may have to adjust your editor window to see the cubes). These cubes are located above the original cube, with relatively large intervals between them. This interval represents the default interval the Clone Object applies. Since we will be creating a surface comprised of many cubes we will now adjust the Cloner Object's settings accordingly.

Select the Cloner Object in the Object Manager and take a look at its editable settings in the Attribute Manager below. You will see that the Cloner Object's Y-value is set to 50m, but we need our clone to move in the Z and X directions. Set Mode from Iterate to Grid Array at the top of the Attribute Manager. The clones will now be arranged into a cube. To flatten our arrangement of clones we will now set the Count's Y-value to 1. Your scene should now look like this:



Our surface is taking shape nicely. Only the number of clones needs to be increased. Set the Cloner Object's X and Z count to 25 each. In order to increase the density of the surface, the clones need to be closer together. To do this, change the Cloner Object's Size X and Z values from 200 to 150. The Y-value can remain unchanged since we haven't cloned the cube in the Y direction. Your scene should now look like this:



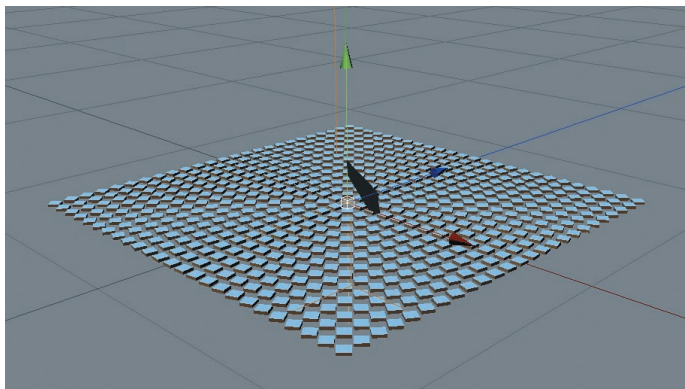
All we need now is a Target Object and a Target Effector. Create a Target Effector by selecting the Cloner Object in the Object Manager and subsequently selecting Target Effector from the MoGraph menu (main menu: MoGraph / Target Effector).

Note:

Always make sure that the Cloner Object is selected when adding an Effector. The Effector will then automatically be added to the Cloner Object's Effector tab. Otherwise you may forget to add the Effector manually and wonder why the Effector is not working. You can read more about Effectors in the integrated CINEMA 4D help files.

By having selected the Cloner Object subsequently assigning the Effector on the Cloner Object's Effectors page is no longer necessary. The Effector already knows it should affect the Cloner Object.

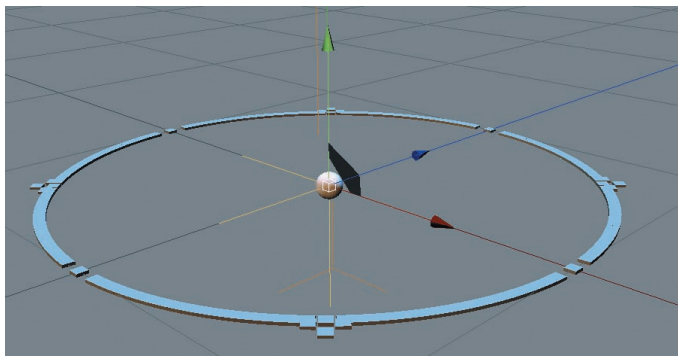
Your scene should now look like this:



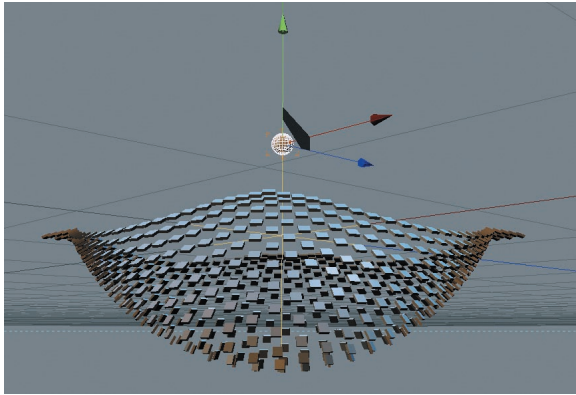
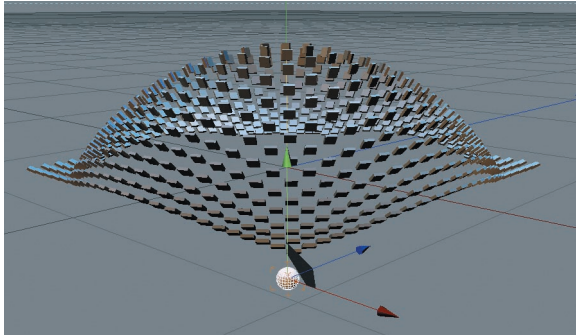
Next we will add a sphere to serve as a Target Object. It really is not necessary to add this sphere but it we will add on here in order to better demonstrate (visually) how this effect works.

Create a sphere and set its radius from 100 to 5 in the Attribute Manager. The sphere now has to be made the Target Object. Open the Target Effector's Effector tab in the Attribute Manager and drag the sphere from the Object Manager into the Effector tab's Target Object text field. That's it!

You can now move the sphere in the editor view and the cube clones will always follow the sphere. We will now go a step further and apply a Repel function, located in the Target Effector's Effector tab. Leave the sphere at the center of the scene (where it was created) and activate the Repel function. The result should look like this:

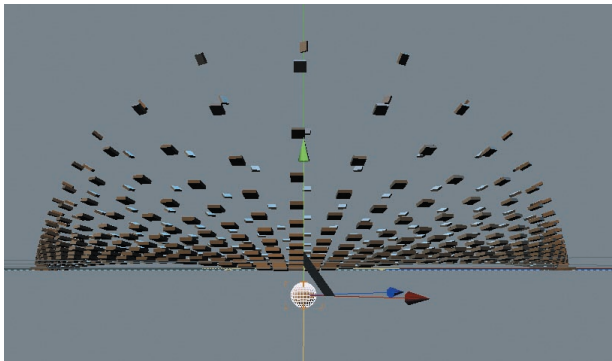


The clones will be repelled radially from the sphere. Move the sphere along any axis and see what effects result:



That basically completes our tutorial, except for the fact that we wanted to simulate the effect shown in the screenshots in chapter 2...

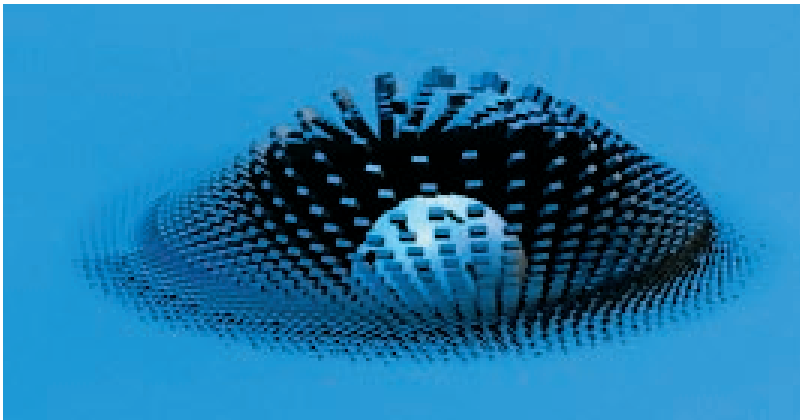
To achieve this effect we will have to add an additional function: Falloff. Place the sphere at the center of and slightly below the clone field.



Open the Target Effector's Falloff tab in the Attribute Manager. Set Shape from Infinite to Sphere and set Scale to 20%. This will define a smaller radius within which our Effector will affect the clones. Our clones are still being repelled much too far away from the sphere. To change this, go to the Target Effector's Effector tab and reduce Distance to 20m and Distance Strength to 50%.

If you move the sphere you will see that only a small number of clones are affected by the movement of the sphere and the remaining clones are not affected at all. In order for all clones to be affected by the movement of the sphere the Target Effector must be made a child of the sphere (Object Manager).

The following look can be achieved by simply increasing the number of clones:



5. Tips & Tricks

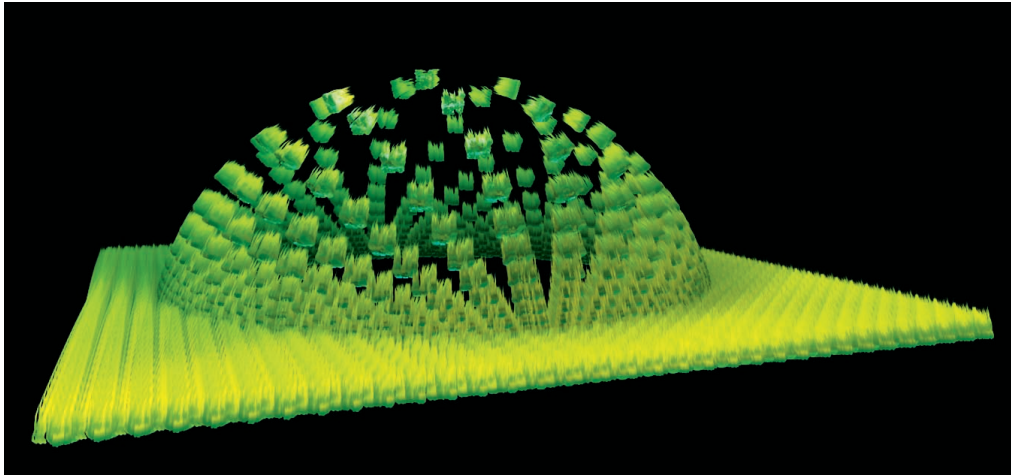
- The MoGraph Cloner Object contains the Object mode. The quantity of clones will orient itself according to the points (default setting) of the object to which they are linked. HyperNURBS can be used to create a smooth animation with a high number of clones. The HyperNURBS object will be used by the Cloner Object in place of an actual polygon object. Depending on the HyperNURBS' settings, the number of clones, for display in the editor view or for rendering, can be controlled using HyperNURBS subdivision. This means you can, for example, have no clones visible in the editor view yet very many visible when the scene is rendered.

- As mentioned in chapter two, the Delay Effector will delay all movement of a Cloner Object's keyed animations. If, for example, you want to morph a face consisting of many objects into another one: Set the Cloner Object's mode to Object and define an object in the Object field. Objects can then be switched per keyframing. Because the resulting animation would simply switch the objects from one state to the next in the next frame, the Delay Effector can be applied so the Cloner Objects transform slowly.

- Unbelievable but true: MoGraph can even be combined with HAIR!

Simply plant hair onto a polygon object, make the HAIR object a child of the polygon object and this in turn a child of the MoGraph Cloner Object. Each animated clone will then be covered with the hair you just created – including HAIR dynamics! Our Quickstart scene could then look something like this.

Now go wild with MoGraph and see what interesting animations you can create. Have fun!



Welcome To 3D Ghosting

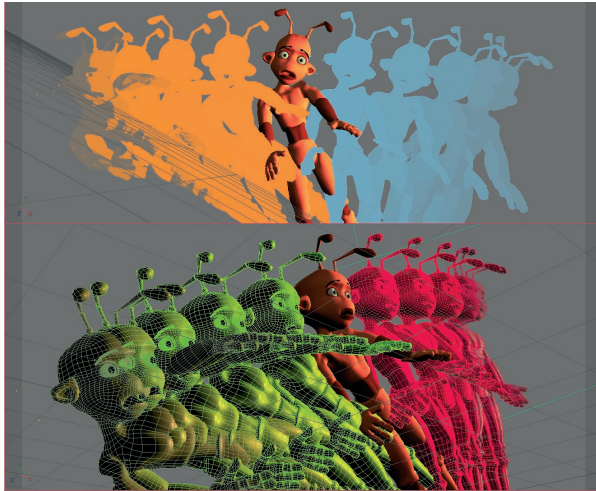
1. Introduction

A very helpful tool for character animators has been added to CINEMA 4D. Especially artists coming from traditional 3D animation will benefit from this important tool, well-known in the field of 2D animation, for fine-tuning animation movement. A 2D animator draws a few frames of animation and lays semi-transparent paper onto a light table so all drawn frames can be viewed at once. Alternatively, the animator flips back-and-forth through the drawn frames in order to get an impression of the animation motion. CINEMA 4D R11 users can, however, save themselves the time, effort and a light bulb or two because of this new CINEMA 4D tool.

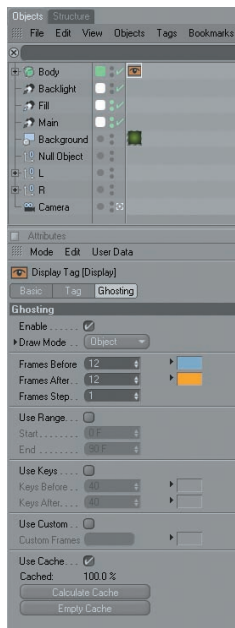
The technique we are talking about is “onion skinning” and the CINEMA 4D tool that provides this function is called “3D Ghosting”. This tool lets you get an impression of your character’s movement prior to and after the current frame (in order to make any necessary corrections) without having to move a slider back and forth. Any animated object’s motion path can be displayed by assigning that object a Display tag. The animator can decide if frames prior to, after or both should be displayed as “ghost” images. Below is a screenshot with examples of 3D Ghosting with default settings.



The frames following the current frame (upcoming frames) are displayed in orange and the frames prior to the current frame (previous frames) are displayed in light blue. The figure not displayed as a wireframe represents the current frame. Numerous settings are available that let you achieve exactly the look you want or need. You can, for example, define the number of frames that should be displayed, how long they should be displayed and how they should be displayed (e.g. color, display mode).



We will show you how this is done with just a few clicks! The next screenshot depicts a typical CINEMA 4D scene:



- Right-click on the desired object and assign it a Display tag (CINEMA 4D Tags/Display)

- Select the Display tag in the Object Manager and then switch to the “Ghosting” tab in the Attribute Manager
- Activate the Enable option and click on the “Calculate Cache” button at the bottom of the Attribute Manager.

That’s it! All you have to do now is fine-tune the Ghosting to fit your needs. For example, you can change the Draw and/or Shading Mode (click on the small arrow next to the Draw Mode parameter); you can use the Frame Before and Frame After values to define how many frames should be displayed before and after the current frame; a Frame Step value of 1 will display every frame. Raising this value to 2 would halve the number of frames, a value of 3 will display only a third of the frames and so on. This gives you an impression of how quickly and easily you can fine-tune Ghosting to suit your needs.

Tips: The way in which HyperNURBS work presents 3D Ghosting with a limitation: If a Display tag has been assigned to a polygon mesh that is a Child of an active HyperNURBS object the Ghosting will not be displayed after the cache has been calculated. To resolve this you can either deactivate the HyperNURBS object before and reactivate it again after calculating the cache or assign the Display tag to the HyperNURBS object in the first place.

3D Ghosting only works on objects that have been animated directly (e.g. a cube animated using Keyframes) without having to calculate the cache. If a mesh has been deformed by a skin a cache will have to be created. However, the cache must not be cleared every time a change is made to the animation. Simply click on “Calculate Cache” and the 3D Ghosting data will be updated.

Welcome to Projection Man

1. Introduction

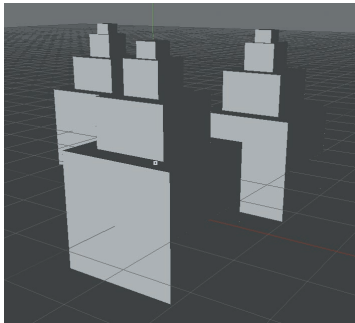
Once you have completed this tutorial you will be able to save a great amount of working time and maybe even create scenes you never would have been able to without this tool. This tutorial is primarily geared towards matte painters but can also be used by any 3D artist to keep from having to texture a great number of objects. For those of you unfamiliar with the term “matte painting”, here is a brief description of what this is: Matte painters mostly work in the movie industry and create (paint) background imagery for movie scenes. These backgrounds are for the most part so realistic that the viewer assumes they are real-world backgrounds. An example of matte painting is a scene in which a king on his horse rides across a virtual landscape that, on the one hand, does not exist in the real world and on the other hand does not have to be built in 3D. The matte painter paints the desired background and the king and his horse are simply composited into the scene.

Advancing technology has also made it possible to create matte paintings in 3D using a computer, which makes it possible to animate a camera and maintain a correct angle of view of the background. This would not be possible using traditional 2D techniques. The disadvantage (if you can call it that) to using 3D matte painting is that a “single image” cannot be used – the scene must be modeled and all objects must be textured. And this is exactly where Projection Man comes in.

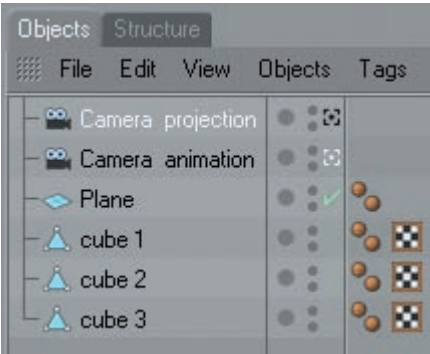
Let’s say you have a scene with a city containing hundreds of buildings. Instead of having to texture each one of these buildings all you have to do is create one or maybe two matte paintings and project it correctly onto the scene. You define the camera’s position and start Projection Man that in turn calculates the location of the geometry and starts Photoshop. In the image that opens in Photoshop you can now paint from the angle of view of that C4D camera. When you have finished painting, save the image in Photoshop and reload it in the corresponding material channel in CINEMA 4D.

Done! Projection Man will now project your painted image onto the geometry of that object (or even several objects) in real-time. Sound complicated? Then let’s work through the following tutorial together and you will see how this tool can free up valuable time for many artists around the world!

Open the file, “QS_PMan_Start.c4d”



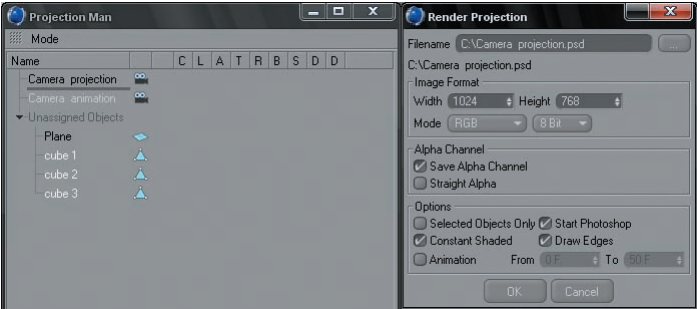
This is a very simplified version of a city scene in which a camera is animated to move in slightly to the buildings. Play the animation once (small green arrow below the Viewport). You can see how the angle of view changes. In traditional matte painting we would have a simple zoom in which the angle of the front building would not change in relation to the others. Our buildings, however, still need to be textured. Each building could be textured individually (which would normally not be much work for just three objects) or you can use Projection Man (e.g. if you had five hundred buildings staring at you waiting to be textured). Our scene contains two cameras. In order for Projection Man to be able to open Photoshop, the correct path to the Photoshop executable file must be entered in CINEMA 4D. Open the C4D Projection Man preferences menu (main menu: Edit / Preferences / Projection Man). Enter (or navigate to) the location of the Photoshop.exe file on your computer. Let's take a closer look at our scene.



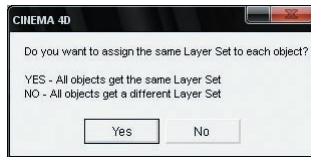
The first camera ("Camera projection") is the camera that Projection Man will use to project a painted image onto the surfaces of the buildings.

The second camera ("Camera animation") is the camera through which we just viewed the animated approach to our buildings. We must now let Projection Man know which objects it should use for the projection. An this is how it's done:

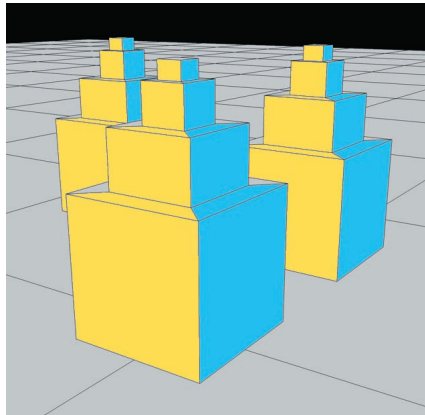
Make sure you return your animation to frame 0. Select "Window / Projection Man" from the main C4D menu. In the window that opens, select all three cube objects and drag them onto the "Camera projection" object above (same window). Select "Coverage Render" from the selection menu that opens.



Enter the location to which you want to render the .psd file and click on “OK”. Confirm the prompt that follows with “Yes”.



CINEMA 4D will now automatically start Photoshop and will open the rendered Projection Man image. You can either start painting in Photoshop right away or create a new layer and create your matte painting. In order to get to know how Projection Man works, edit your own image to look like the one below. Of course you can use your favorite colors if you want.

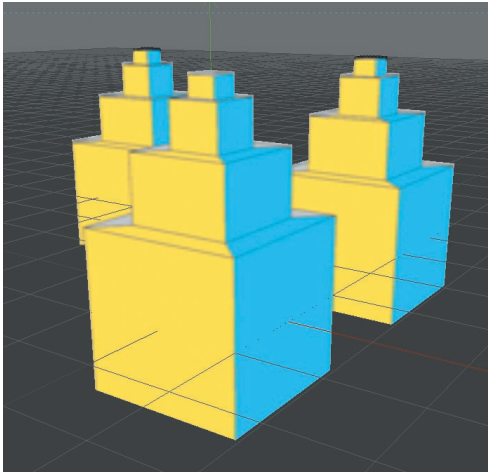


After you have finished modifying your image in Photoshop, save the image. Use the current name and location – otherwise CINEMA 4D will not be able to locate the image!

Return to your Projection Man scene in CINEMA 4D. In the Material Manager, at the bottom left of your interface, you can see that Projection Man has automatically created a new material. Double-click the material and switch to the Luminance channel in the window that opens.

Tip: Projection Man creates the texture automatically in the Luminance channel. This ensures that the scene remains completely unaffected by lights. After all, the scene is supposed to assume the color and brightness traits of your painted texture. This, however, can be changed by either deactivating the material’s Luminance channel and loading the .psd file into the Color channel or by changing the Projection Man default settings in the CINEMA 4D preferences menu so the .psd file is automatically loaded into the Color channel.

We are now in the material’s Luminance channel. Click on the small triangle next to the Texture parameter and select Reload Image. This updates the texture and includes the changes we just made in Photoshop. Now close the Material Manager and your scene should look like this:

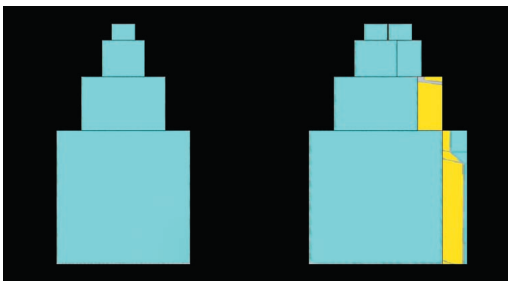


Play the animation. As you can see, Projection Man projects the texture correctly onto all three buildings throughout the animation – and that without having to texture each individual object.

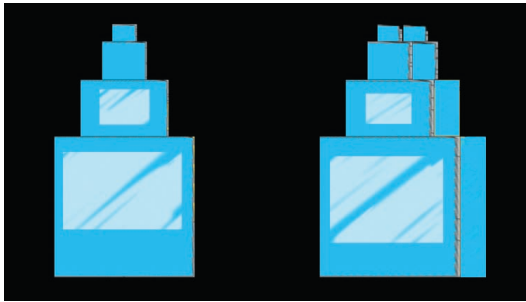
Now we will take Projection Man a step further and add a few windows to the side of our buildings. Normally, painting windows onto the surfaces at these angles would be quite difficult but all we have to do here is add a camera to project the desired image information onto the correct surfaces.

We will point this camera frontally at the light blue surfaces (side view). To create the camera, switch the Viewport to the Right view (Viewport menu: Cameras / Right) and center the view if needed. Create a new camera (main menu: Objects / Scene / Camera). Stay in this view and rename the camera “Camera right” in the Object Manager (double-click on its name). Switch to this camera’s view by clicking on the “plus” symbol next to its name.

Again open the Projection Man window (main menu: Window / Projection Man) and drag “Cube 1” and Cube 2” onto “Camera right”. Select “Coverage Render” and confirm all prompts with “Yes”. The newly rendered image will also be opened in Photoshop.



Edit the image to look (more or less) like the one below:



Save the Photoshop file, and return to CINEMA 4D. Again a new material has been created, this time with the name, “PMat Camera right”. Double-click the material’s icon and reload the image in the Luminance channel. The texture will be updated in the Viewport immediately and the windows are projected correctly onto the objects. Play the animation.

You now know how easy it is to texture a scene using Projection Man, even without prior knowledge of 3D texturing. If needed, more cameras can be added and used for projection in order to create longer and more complex camera animations or to compensate for areas that may not have been mapped by another camera. As you saw in our example, all you need for a simple zoom animation is a single view and a single “painting”. If the camera were to move from left to right you would most likely have to create a matte painting for the start and end positions of the animated camera. In any event, Projection Man saves you from having to texture all 15,876 buildings in a single city!

Welcome to Non-Linear Animation in the Timeline

Non-linear animation lets you combine recorded data using motion clips or MoCap animation layers, or manually. The motion layer method displays animation data as individual “motion clips” and requires the animation data to be mixed horizontally. These motion clips can be placed successively or be superimposed (laid on top of each other) in order to create a transitional morph from one animation to the next. The originally recorded keys will be removed and saved in the corresponding “motion source” file.

The Animation Layer System is different. Individual layers are ordered vertically and animations are recorded one after the other on individual layers. The layer that was active while a particular key was being recorded will contain that key's animation. This lets you activate or deactivate animation layers, similarly to sound tracks in an audio editing program. The active layers will be mixed, which offers a myriad of possibilities and can save a great deal of time. You can, for example, duplicate a layer and modify the animation contained on it without affecting the original animation. You can keep both layers and later decide which animation stays and which one goes. If this sounds a little complicated for you, you can either refer to the beginning of the Quickstart Manual where all new features are described in detail or you can get started with our tutorial.

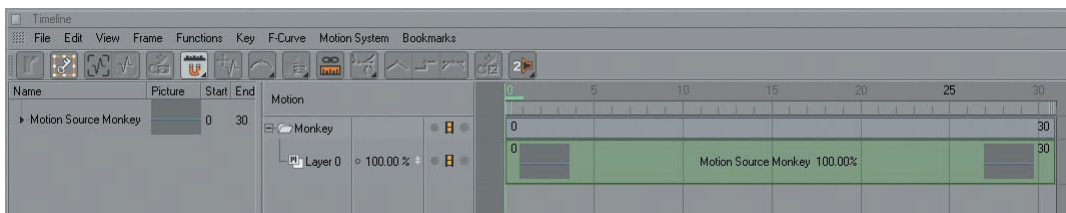
1. Motion Layer

We will now explore the motion layer method in more detail. Open the file, “QS_ML_Start.c4d” and switch the CINEMA 4D layout to Animation (top icon in the left Icon Palette, directly below the Undo button). As you can see, our scene contains a monkey head. We will use this head to create a simple character animation. In the course of this tutorial we will assign two simple animations to the monkey head: Forward and backward nodding motion; up-and-down motion. You may wonder why a monkey would ever nod or move its head up and down. Surely, monkeys that only move their heads back and forth or up and down are extremely rare in the wild (or even extinct) but we thought a very simple, unrealistic animation would be a better way to demonstrate the basic principle of non-linear animation (NLA) than having King-Kong push his way through the jungle.

When you play the animation you will see that we have already prepared an animation for you. The monkey head moves from left to right. Stop the animation and select the “Monkey” Null Object.

Motion layers work together with all recorded keys within a given hierarchy. Simply select the top object in the hierarchy and all its keys will be included (regardless if it's a character's controller or animated geometry).

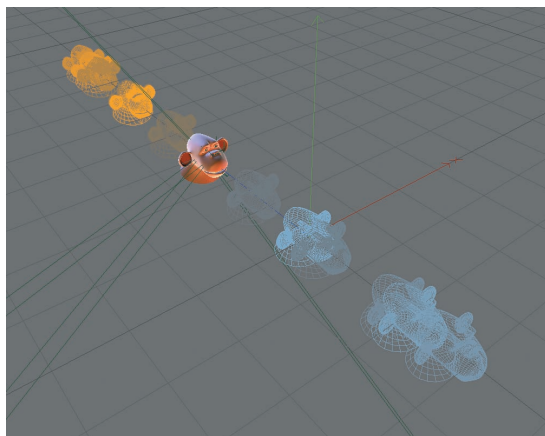
In the C4D main menu, select the Add Motion Clip... function from the Animation menu. Confirm the window that opens without making any changes. As you can see, our Timeline now looks different. A motion source (left) and a motion clip (right) of the Null Object's (and its Children) animation were created.



The original Keyframes were deleted. If we play the animation again we will see that the left-to-right animation nevertheless still exists. If we click on the “Monkey” Null Object, though, we will see that no Keyframes exist for this object. These Keyframes are now located in the motion source that was linked to the Motion Clip. You can now use the same motion source for more than one Motion Clip. Note that CINEMA 4D automatically switched to Motion Mode (Timeline menu: View / Motion Mode) when the Motion Clip was added. If you want to modify the animation contained in this Motion Clip you will have to switch to Key Mode first (Timeline menu: View / Key Mode). Here you will find the motion source that can be modified as desired. Since the original Keyframes for the “Monkey” Null Object no longer exist we can record a new animation and add it as a new source and clip. The motion system must be deactivated before this is done. With “motion system” we mean deactivating the complete motion system, not just switching from Motion Mode to Key Mode!

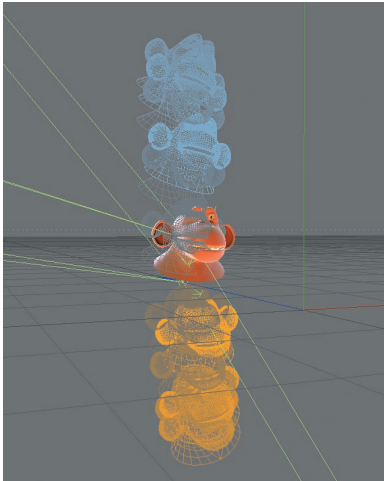
This is done by selecting Edit / Use Motion System in the main CINEMA 4D menu. If the icon has a dark background it is not active – a light background indicates that it is active. If you cannot see the “Monkey” Null Object in the left part of the track manager, select Bookmarks / Default Bookmark from the Timeline menu. If the Null Object is still selected, set a key at frame “0” (click on the “key” icon above the Timeline to set a key). Go to frame 7 (drag the green grabber below the Viewport to frame 7) and move the monkey head a little to the front (along the Z-axis) in the Viewport. Set another Keyframe.

Go to frame 15, move the monkey head back along the Z-axis. Set another Keyframe. Now go to frame 30 and return the head to its original position. This can be done easily by copying frame 0 to frame 30. All you have to do is simultaneously press the Ctrl key while dragging frame 0 onto frame 30. Your animation should look similar to the image below (here with 3D Ghosting):



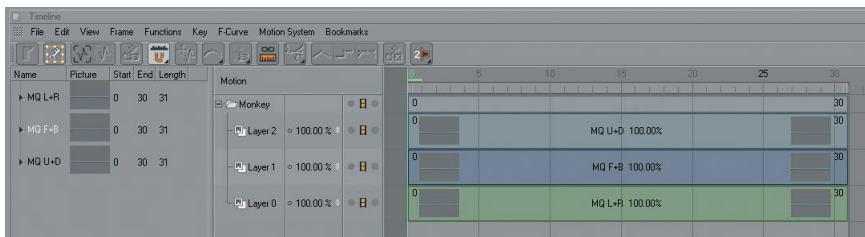
Again select Animation / Add Motion Clip from the C4D main menu. Another motion source has been added and we are again in Motion Mode. We will now add another animation.

Our motion system is still deactivated, which lets us begin directly with the setup of your third animation. Set a few Keyframes for an up-and-down motion of the head, using the same method as for the previous animation (by moving the slider to a desired frame, moving the head, setting a Keyframe, etc.). Your up-and-down movement should look (at least somewhat) like the image below:



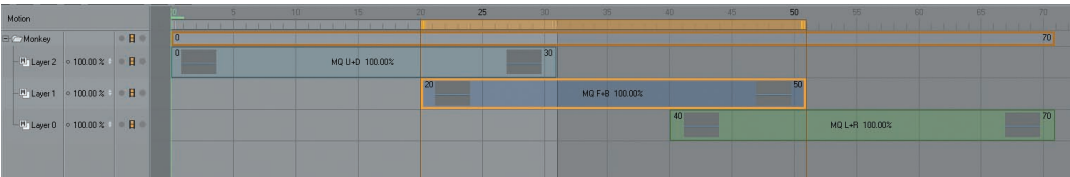
Create another motion source with an automatically created Motion Clip (you already know how this is done). Our Timeline now contains three Motion Clips.

For a better overview, rename the sources (double-click on each name) from top to bottom, “MQ U+D”, “MQ F+B” and “MQ L+R”, respectively. This will give you a better overview which source and which Motion Clip contain which animation.



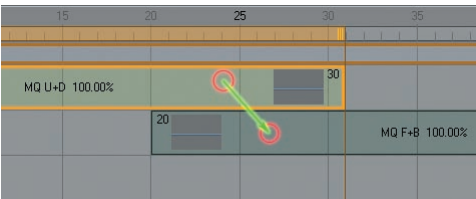
You can also get a better overview by adding an image or even a movie for each motion source. Right-clicking on a motion source (at the left of the window) will open a Context Menu. Select the Add Picture to Motion Source command and select any image or movie format. This is especially helpful for cleaning up the overview when several artists are working together on a large project.

Now move the cursor over one of the Motion Clips in the Timeline. The cursor will either turn into a double arrow with which you can move the Clip or an orange box with large “+” in it. The latter appears when the cursor is placed over the beginning or end of a Clip and lets you scale (shorten or extend) the Clip. If a Clip is shortened the animation will play faster, i.e. the motion will take place within a fewer number of frames. Move and scale the Clips in your scene as pictured below – this will allow us to blend from one clip to the next.

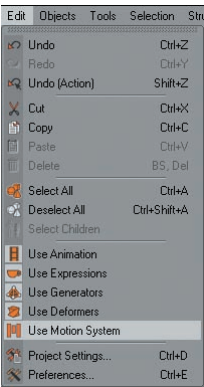


There are two ways in which to mix Clips. They can either remain as independent layers, for which transitions must be created or they can be dragged onto each other on a single layer.

For the first method, right-click on a Motion Clip and select the Make Transition command from the Context Menu that appears. Now click and drag a connecting line from the top Clip to the Clip below it.



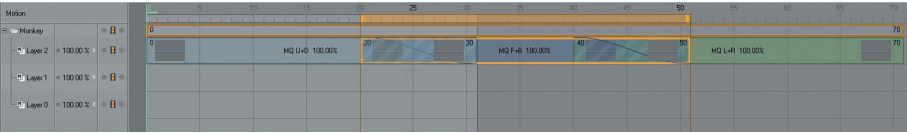
This creates a soft transition. Activate the Motion System (main menu: Edit / Use Motion System) to see how the transition looks.



Play the animation and you will see that the animation, MQ U+D, now gradually transitions into the MQ F+B animation. Select the Undo option to undo the transition you just created. We will now turn our attention to the second method for mixing Motion Clips.

If a transition already exists is denoted by small red circles at the left next to the percentage numbers. These circles indicate that Keyframes were recorded for the transition from one Clip to the next. Undo all previous actions until these red circles are gone. This way you can be sure no keys exist for this transition for our next step – laying them on top of each other.

Place the cursor over the center Motion Clip. Press the Shift button on your keyboard and drag this Clip up into the first Clip's layer. Do the same with the Clip at the right.



Play the animation. All three Clips will now transition smoothly. The advantage of the first mixing method, however, is that each Clip lies on its own layer and can be turned on or off separately (click on the orange and black film strip symbol in the Layer Manager). The Motion Layer system offers innumerable possibilities for creating transition between animation data. The fact that individual motion sources can be saved as presets you can create your own animation library that you can later use for other characters or objects. Play around with the system a little, including the Motion System tag's parameters to get to know the system in more detail.

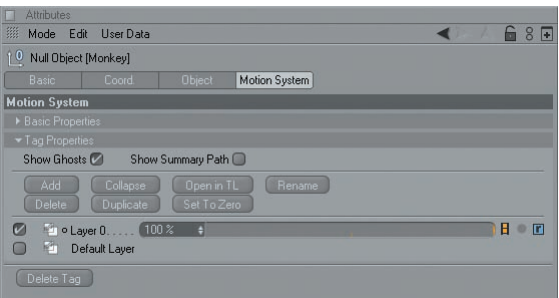
Attention!!

Never mix up the Motion Source and Motion Clips functions. The Motion Sources are located at the left of the Timeline and the Motion Clips are located in the Timeline itself. The latter are “merely” links to the original sources. This means that the source contains the actual animation data and the Motion Clips (with which the animations are mixed and can be stretched or expanded in the Timeline) simply reference this source but themselves contain no animation data. This can also be seen in the Attribute Manager after selecting a Motion Clip. The Attribute Manager contains a “Source” field in which the source for that particular Motion Clip is displayed. This way, several Motion Clips can use the same source (non-destructive animation).

2. Animation Layer

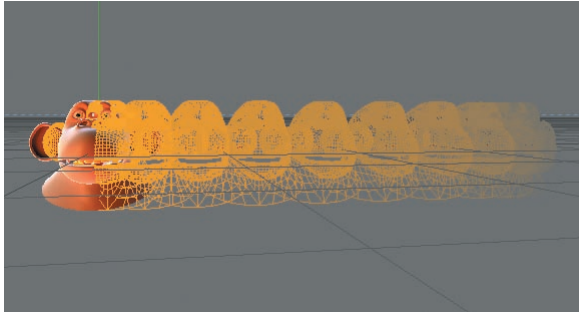
The Animation Layer System is a powerful yet easy to learn tool for managing and mixing animation using layers.

Open the file “QS_AL_Start.c4d”. Again we see our trusted monkey head, however without any animation applied to it. And again we will animate the monkey head with typical monkey-like movements. If you haven’t already done so, switch to the predefined “Animation” layout. Select the “Monkey” Null Object and select Animation / Add Animation Layer from the C4D main menu. Our Null Object has just been assigned a Motion tag. In the Attribute Manager’s Motion System tab we can see that the current layer is “0”.

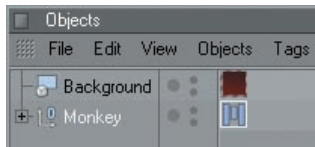


The check mark at the left of the Level 0 parameter indicates that layer 0 is the active layer.

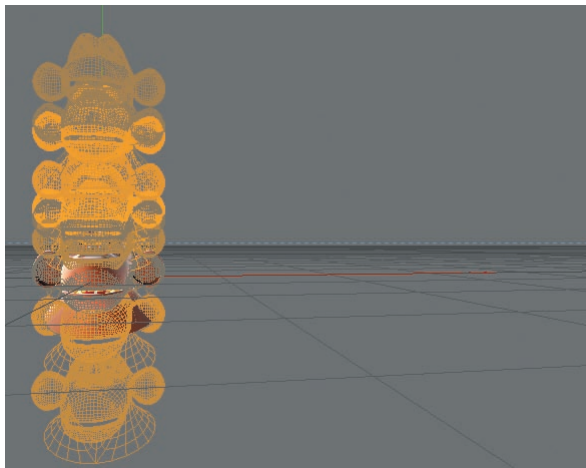
You can begin animating right away. Create a couple of simple movements by moving the slider to a desired frame, moving the head and saving a keyframe. Although how the monkey head moves is irrelevant we suggest you use a standard left-to-right movement for now.



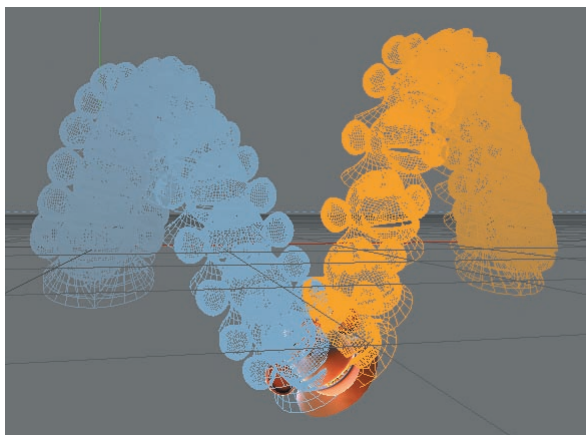
Once you have finished setting all your keyframes click on the “Add” button in the Attribute Manager. If you do not see this button it is probably because the Motion tag or Null Object are not selected. Simply select either the “Monkey” Null Object or its blue Motion tag.



If you have done everything correctly a Layer 1 should have been added after clicking on the Add button. Layer 1 will automatically be made the active layer. You will also notice that the keyframes we just set are no longer visible but the monkey head still moves in accordance to these if the animation is played. This is due to the fact that, in contrast to the Motion System, the Animation Layer System adds keys with each layer without deleting the previously created keys. In our case the previously created keys are not visible because they belong to Layer 0, which is not currently active. Create a few new movements (of course with new keyframes) and play the new animation.



Both layers will now be mixed. You can add any number of additional layers. Each time the keyframes will belong to the layer that was active when they were created. After setting keyframes on three separate layers we will have three separate animations that will be mixed. If, for example, we created a left-to-right movement on Layer 0, an up-and-down movement on Layer 1 and a left-to-right shaking of the head on Layer 3 the animation could look like this:



Layers can also be turned on or off (click on the orange and black film strip in the Attribute Manager) or copied. A layer can be copied if you want to make modifications to an existing animation without modifying the original animation. We hope you see the advantages the Animation Layer System has to offer. Keyframes no longer have to be gathered on a “large layer”, something that can cause you to have a poor overview of your animation.

Once you’ve worked with the system a little you will realize that it is an indispensable tool for your animation needs. Your ingenuity and imagination are this tool’s only limitations.

We hope you enjoy all the new features and functionalities CINEMA 4D R11 has to offer. Have fun bringing your imagination to life!



