CINEMA 4D

Programming Team
Reinhard Hintzenstern, Tilo Kühn, Thomas Kunert, Richard Kurz,
Christian Losch, Philip Losch, David O’Reilly

Additional Programming
Wilfried Behne, Sven Behne, Michael Breitzke, Kiril Dinev, David Farmer, Jan Eric Hoffmann,
Jamie Halmick, Nina Ivanova, Markus Jakubietz, Eduardo Olivaures, Hendrik Steffen,
 Eric Sommerlade, Jens Uhlig, Thomas Zeier,

Writers
Oliver Becker, Arndt von Koenigsmarck, David Link, Stephen Marriott, Matthew O’Neill,
David O’Reilly, Janine Pauke, Perry Stacy, Jeff Walker

Layout
Heike Bauer, Harald Egel, Jeff Walker

Translation
Oliver Becker, Michael Giebel, Arno Löwecke, Björn Mar, Janine Pauke, Luke Stacy, Marco Tillmann

Cover Graphic
Heike Bauer, Onur Pekdemir

Special Thanks
Kevin Aguirre, Phil ‘Captain 3D’ McNally, NAAM, Kai Pedersen,
Christian Rambow, Holger Schlömann, Bunk Timmer

Tutorial Manual Copyright © 2001-2002 by MAXON Computer Ltd. 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, CINEMA 4D, Hyper NURBS, and 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, and Windows NT are either registered trademarks or trademarks of Microsoft Corporation in the U.S. and/or other countries. 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.
MAXON Computer End User License Agreement

NOTICE TO USER

WITH THE INSTALLATION OF CINEMA 4D (THE “SOFTWARE”) A CONTRACT IS CONCLUDED BETWEEN YOU (“YOU” OR THE “USER”) AND MAXON COMPUTER GMBH (THE “LICENSOR”), A COMPANY UNDER GERMAN LAW WITH RESIDENCE IN FRIEDRICHSDORF, GERMANY.

WHEREAS BY USING AND/OR INSTALLING THE SOFTWARE YOU ACCEPT ALL THE TERMS AND CONDITIONS OF THIS AGREEMENT. IN THE CASE OF NON-ACCEPTANCE OF THIS LICENSE YOU ARE NOT PERMITTED TO INSTALL THE SOFTWARE.

IF YOU DO NOT ACCEPT THIS LICENSE PLEASE SEND THE SOFTWARE TOGETHER WITH ACCOMPANYING DOCUMENTATION TO MAXON COMPUTER OR TO THE SUPPLIER WHERE YOU BOUGHT THE SOFTWARE.

1. General

Under this contract the Licensor grants to you, the User, a non-exclusive license to use the Software and its associated documentation. The Software itself, as well as the copy of the Software or any other copy you are authorized to make under this license, remain the property of the Licensor.

2. Use of the Software

You are authorized to copy the Software as far as the copy is necessary to use the Software. Necessary copies are the installation of the program from the original disk to the mass storage medium of your hardware as well as the loading of the program into RAM.

(2) Furthermore the User is entitled to make a backup copy. However only one backup copy may be made and kept in store. This backup copy must be identified as a backup copy of the licensed Software.

(3) Further copies are not permitted; this also includes the making of a hard copy of the program code on a printer as well as copies, in any form, of the documentation.

3. Multiple use and network operation

(1) You may use the Software on any single hardware platform, Macintosh or Windows, and must decide on the platform (Macintosh or Windows operating system) at the time of installation of the Software. If you change the hardware you are obliged to delete the Software from the mass storage medium of the hardware used up to then. A simultaneous installation or use on more than one hardware system is not permitted.

(2) The use of the licensed Software for network operation or other client server systems is prohibited if this opens the possibility of simultaneous multiple use of the Software. In the case that you intend to use the Software within a network or other client server system you should ensure that multiple use is not possible by employing the necessary access security. Otherwise you will be required to pay to the Licensor a special network license fee, the amount of which is determined by the number of Users admitted to the network.

(3) The license fee for network operation of the Software will be communicated to you by the Licensor immediately after you have indicated the number of admitted users in writing. The correct address of the Licensor is given in the manual and also at the end of this contract. The network use may start only after the relevant license fee is completely paid.
4. Transfer

(1) You may not rent, lease, sublicense or lend the Software or documentation. You may, however, transfer all your rights to use the Software to another person or legal entity provided that you transfer this agreement, the Software, including all copies, updates or prior versions as well as all documentation to such person or entity and that you retain no copies, including copies stored on a computer and that the other person agrees that the terms of this agreement remain valid and that his acceptance is communicated to the Licensor.

(2) You are obliged to carefully store the terms of the agreement. Prior to the transfer of the Software you should inform the new user of these terms. In the case that the new user does not have the terms at hand at the time of the transfer of the Software, he is obliged to request a second copy from the Licensor, the cost of which is born by the new licensee.

(3) After transfer of this license to another user you no longer have a license to use the Software.

5. Updates

If the Software is an update to a previous version of the Software, you must possess a valid licence to such previous version in order to use the update. You may continue to use the previous version of the Software only to help the transition to and the installation of the update. After 90 days from the receipt of the update your licence for the previous version of the Software expires and you are no longer permitted to use the previous version of the Software, except as necessary to install the update.

6. Recompilation and changes of the Software

(1) The recompilation of the provided program code into other code forms as well as all other types of reverse engineering of the different phases of Software production including any alterations of the Software are strictly not allowed.

(2) The removal of the security against copy or similar safety system is only permitted if a faultless performance of the Software is impaired or hindered by such security. The burden of proof for the fact that the performance of the program is impaired or hindered by the security device rests with the User.

(3) Copyright notices, serial numbers or other identifications of the Software may not be removed or changed. The Software is owned by the Licensor and its structure, organization and code are the valuable trade secrets of the Licensor. It is also protected by United States Copyright and International Treaty provisions. Except as stated above, this agreement does not grant you any intellectual property rights on the Software.

7. Limited warranty

(1) The parties to this agreement hereby agree that at present it is not possible to develop and produce software in such a way that it is fit for any conditions of use without problems. The Licensor warrants that the Software will perform substantially in accordance with the documentation. The Licensor does not warrant that the Software and the documentation comply with certain requirements and purposes of the User or works together with other software used by the licensee. You are obliged to check the Software and the documentation carefully immediately upon receipt and inform the Licensor in writing of apparent defects 14 days after receipt. Latent defects have to be communicated in the same manner immediately after their discovery. Otherwise the Software and documentation are considered to be faultless. The defects, in particular the symptoms that occurred, are to be described in detail in as much as you are able to do so. The warranty is granted for a period of 6 months from delivery of the Software (for the date of which the date of the purchase according to the invoice is decisive). The Licensor is free to cure the defects by free repair or provision of a faultless update.

(2) The Licensor and its suppliers do not and cannot warrant the performance and the results you may obtain by using the Software or documentation. The foregoing states the sole and exclusive remedies for the Licensor’s or its suppliers’ breach of
warranty, except for the foregoing limited warranty. The Licensor and its suppliers make no warranties, express or implied, as to noninfringement of third party rights, merchantability, or fitness for any particular purpose. In no event will the Licensor or its suppliers be liable for any consequential, incidental or special damages, including any lost profits or lost savings, even if a representative of the Licensor has been advised of the possibility of such damages or for any claim by any third party.

(3) Some states or jurisdictions do not allow the exclusion or limitation of incidental, consequential or special damages, or the exclusion of implied warranties or limitations on how long an implied warranty may last, so the above limitations may not apply to you. In this case a special limited warranty is attached as exhibit to this agreement, which becomes part of this agreement. To the extent permissible, any implied warranties are limited to 6 months. This warranty gives you specific legal rights. You may have other rights which vary from state to state or jurisdiction to jurisdiction. In the case that no special warranty is attached to your contract please contact the Licensor for further warranty information.

The user is obliged to immediately inform the transport agent in writing of any eventual damages in transit and has to provide the licensor with a copy of said correspondence, since all transportation is insured by the licensor if shipment was procured by him.

8. Damage in transit

You are obliged to immediately inform the transport agent in writing of any eventual damages in transit and you should provide the Licensor with a copy of said correspondence, since all transportation is insured by the Licensor if shipment was procured by him.

9. Secrecy

You are obliged to take careful measures to protect the Software and its documentation, in particular the serial number, from access by third parties. You are not permitted to duplicate or pass on the Software or documentation. These obligations apply equally to your employees or other persons engaged by you to operate the programs. You must pass on these obligations to such persons. You are liable for damages in all instances where these obligations have not been met. These obligations apply equally to your employees or other persons he entrusts to use the Software. The User will pass on these obligations to such persons. You are liable to pay the Licensor all damages arising from failure to abide by these terms.

10. Information

In case of transfer of the Software you are obliged to inform the Licensor of the name and full address of the transferee in writing. The address of the Licensor is stated in the manual and at the end of this contract.

11. Data Protection

For the purpose of customer registration and control of proper use of the programs the Licensor will store personal data of the Users in accordance with the German law on Data Protection (Bundesdatenschutzgesetz). This data may only be used for the above-mentioned purposes and will not be accessible to third parties. Upon request of the User the Licensor will at any time inform the User of the data stored with regard to him.

12. Other

(1) This contract includes all rights and obligations of the parties. There are no other agreements. Any changes or alterations of this agreement have to be performed in writing with reference to this agreement and have to be signed by both contracting parties. This also applies to the agreement on abolition of the written form.
(2) This agreement is governed by German law. Place of jurisdiction is the competent court in Frankfurt am Main. This agreement will not be governed by the United Nations Convention on Contracts for the International Sale of Goods, the application of which is expressly excluded.

(3) If any part of this agreement is found void and unenforceable, it will not affect the validity of the balance of the agreement which shall remain valid and enforceable according to its terms.

13. Termination
This agreement shall automatically terminate upon failure by you to comply with its terms despite being given an additional period to do so. In case of termination due to the aforementioned reason, you are obliged to return the program and all documentation to the Licensor. Furthermore, upon request of Licensor you must submit written declaration that you are not in possession of any copy of the Software on data storage devices or on the computer itself.

14. Information and Notices
Should you have any questions concerning this agreement or if you desire to contact MAXON Computer for any reason and for all notifications to be performed under this agreement, please write to:

MAXON Computer GmbH
Max-Planck-Str. 20
D-61381, Friedrichsdorf
Germany

or for North and South America to:

MAXON Computer, Inc.
2640 Lavery Court Suite A
Newbury Park, CA 91320
USA

or for the United Kingdom and Republic of Ireland to:

MAXON Computer Ltd
The Old School, Greenfield
Bedford MK45 5DE
United Kingdom

We will also be pleased to provide you with the address of your nearest supplier.
Introduction

Here is information on how this manual works and how to get the best out of it.

If you’re eager to get into the program, we recommend you jump forward a few pages to the Quickstart tutorials. These will get you going with all of the CINEMA 4D essentials.

This tutorial manual has been specially designed to work with a broad range of learning methods. If you are the kind of person who just wants to get through it all to learn where the important parts are, you may just follow the actions in bold text:

- **Following these steps is the shortest route to achieving the final result.**

  If you would like to know more about an action you’ve just performed, the text underneath the action will go into more detail.

After the Quickstart tutorials you will find specific tutorials for the various stages of the 3D production process: modeling the object, lighting and texturing, animation and rendering.

Still artists may choose to skip the animation features entirely as they might not be interested in the animation elements of the package at this time. Similarly, if you make your money by texturing or modeling objects, you might never need to know about rendering. At the end of the manual there is a Tips and Tricks section which you should find invaluable.

While working through the tutorials you should always keep in mind that there are many ways to accomplish a task in 3D. In some cases we have chosen to use a method to accomplish something that perhaps isn’t the best or most efficient method, but it illustrates the use of a particular tool. Your ultimate goal will be to learn the tools and figure out what works best for you.

Don’t forget to check our web site on a regular basis (www.maxon.net) as we are always adding tutorials for new techniques and solutions to production problems. If you experience any difficulties using CINEMA 4D, please contact our support team through the online support form. We will always do our best to help.

**Upgraders**

If you have used a previous version of CINEMA 4D, you will have a few new working methods to learn because of new tools and features. The main two changes you will notice are the Attributes manager and multiple selections.

Most of the dialogs have been replaced in Release 8 with a single window called the Attributes manager, this enables you to adjust any setting without having to open a different window. It also means all of the settings are displayed in a similar way to one another rather than having lots of unique windows.

Multiple selection means that many of the commands prior to Release 8 which required you to first group the objects you wanted to use have been replaced with the ability to simply select whichever objects you like, no matter where in the hierarchy they are.
Overview

Here we’ll take a look at the different processes involved in 3D work to give us a taste of what we will learn in this tutorial.

Being a successful 3D artist means utilizing a variety of disciplines: painting, sculpting, architecture, engineering, architectural or stage lighting techniques, photography, cinematography, choreography, acting, mathematics/physics, sound design and others. You don’t have to be an expert in all of these areas, but having an understanding of one or a few is helpful.

There are five core techniques when working in 3D. The freelance artist working on his or her own will need to become at least proficient in all five areas. In a design firm of three or more 3D artists, some may be better in one area or another, but it is still best to have an understanding of all. There are few environments where an artist will only work in one specific discipline. The five areas of 3D are:

- Modeling: Building the objects that will be in your scene.
- Texturing: Defining the surface properties of all objects.
- Lighting: Adding illumination, like lighting a stage for a theatre production.
- Animating: Creating motion with keyframes.
- Rendering: Producing the final picture or movie.
There are also related elements which can be essential:

- Visual Effects: Explosions, Melting, etc.
- Sound: Music, Effects or Voice.
- Post Production and/or Editing.

**Modeling**
The act of observation and visual study is the most important skill. Without a discerning eye you cannot even begin to recreate the world around you. To model an object you have to look at the world as a sculptor, architect or engineer. You have to examine an object in terms of how it is structured, the shapes and parts used to define it, and how you might go about building it in the most efficient manner.

**Texturing**
Adding materials to an object means knowing how to define how an object looks using a variety of surface elements such as color, bump, reflection and transparency. Not just, “is it blue or red?” but understanding the textures of an object, and how its surface reacts to light. Not only making it reflective, but also defining the strength of the reflection. Not just making something transparent, but how refractive the transparency, thus changing the way objects appear behind it. These are the skills of a fine artist.

**Lighting**
The best lighting in animation is almost subliminal; present but not obtrusive. Lighting should enhance what you’ve done with the materials and Modeling in your scene. It also creates the mood of your piece. Some talented 3D artists have studied architectural lighting, but lighting skills can also be learned through studying photography and cinematography.

**Animating**
Animating is the most complex task in 3D. It requires the skills of an actor, mathematician, physicist and choreographer. Creating smooth and logical motion can take twice as long as all the other elements of 3D combined. Creating realistic character animation increases that complexity 10-fold.

Animation is the illusion of movement created by flipping through a sequence of still images. As the images quickly replace each other on the screen, it creates the illusion that the figures are moving. Each medium flashes the pictures before our eyes at a specific rate — 24 frames per second for film, for example, and 29.97 frames per second for US television.
In traditional 2D animation each and every image is hand drawn in order to portray movement. The artist pictures the movement in his mind and plans out each drawing to create that action. As each picture is drawn, movement is shown through the change of the objects in the scene and their positions in the drawing. The number of images and the rate at which they are viewed controls the speed of the action. A technique called stop motion is used for animating puppets, clay (aka claymation), cutouts, and sand or paint. The animation is created by recording each frame, one-by-one, while moving or changing the elements in the scene. The result is the illusion of fluid change or movement.

In 3D animation, models are created in the 3D environment, materials are applied and lighting is added to the scene. To create motion, the artist has only to create the key moments or ‘keyframes’ of action, and the software creates or interpolates the movement between the keyframes. Ultimately, this 3D action is rendered to 2D images, which are played back in sequence to create the illusion of movement.

3D animation gives you the ability to be producer, scriptwriter, director, actor, set designer, lighting technician, editor and more all rolled into one.

**Rendering**

Rendering is where all of our hard work comes together to form a complete image. Once we’ve hit the Render button it is time for the computer to do the hard work while we sit back and relax. During rendering the computer draws an image of our 3D scene while taking into account all the creations that are in that scene.

Several aspects should be considered when rendering:

Image quality — Various render settings will increase image quality, these include better antialiasing for smoother diagonal lines and deeper rays for more complicated reflections. In general, due to the number of calculations required to draw the scene, increasing image quality increases render times. Ideally, keep settings as low as acceptable to ensure fast render times.

Lighting — Different types of lighting can dramatically increase render times. Be careful when using advanced features like radiosity and area lights as render times can be high. Various adjustments can be made to ensure good results without massive render times, some of which are explored in the Rendering and Tips and Tricks chapters of this tutorial manual.

Output size — We can render a scene up to a size of 16,000 by 16,000 pixels. Naturally, higher resolutions increase render times so only render at the size needed. Also note that certain effects, like caustics for example, will take the same time to calculate regardless of output size. In this case a lot of render time will be dedicated to the calculation of the caustics, after which the output size will be taken into account.
File format — Remember to choose the format when rendering. CINEMA 4D supports many 2D and 3D formats. Remember that some formats compress image quality, so we should pick a format that best suits our needs. A good tip is to save from CINEMA 4D with little or no compression and then use other software to compress the files if need be.

Experience is the best way to learn about rendering. Every 3D scene is different and render settings that work well for one scene may not be the optimum for another. Take time to experiment and perform small test renders before outputting any large image.

Summary

No one ever asks how long it takes to become an artist or how long it takes to become a proficient musician, but people always seem to want to know how long it will take to become a 3D artist. The real question is, how long will it take to become competent in the program so you can create the type of image or animation you want to do. The answer depends on the skill set of each individual and the type of work you choose to do.

It is worth noting that there is a difference between someone who is more technician and someone who is more traditional artist. Those who are technically inclined and have experience with other digital graphics tools tend to pick up a 3D program more quickly. However, the technician also tends to have less artistic training. They are able to move through the program and complete projects faster, but tend to struggle more to achieve a specific look or style. If you are more technically inclined, you should make an effort to learn more traditional artistic skills. For the artist trained in traditional methods, digital tools are a new challenge. It tends to take someone relatively new to digital tools a bit longer to understand the concept. However, once learned, the artist shows a greater ability to deliver realistic looking imagery and more imaginative animation. If you are a classically trained artist, it would be to your benefit to have a better understanding of digital tools and how you can use them to achieve your desired result.

Best of all is to be a traditionally trained artist who is technically inclined. This person is able to master the technical challenges while still maintaining a critical eye towards the finished artistic needs of the project. The bottom-line is, even the world’s best animators are challenged every day by what the available tools can and cannot do. Any of them will tell you they are still learning and have much more to learn.

In short, learning CINEMA 4D can take two weeks or three months depending on your skill set and background in graphics programs. Mastering 3D takes a lifetime.
About the Interface

If you read only one section in this tutorial, let it be this one. Finding your way around the CINEMA 4D interface is an essential prerequisite.

The default CINEMA 4D interface consists of a large editor window or ‘viewport’ which is surrounded by icon palettes and several manager windows. The icon palettes are the strips of images we see along the edges of the screen, the managers are the larger areas where information will be presented.

- Choose File > Open from the main menu and load the file named ‘quickstart.c4d’ from the Quickstart folder in the CINEMA 4D Tutorials folder.

By loading this example scene it is easier for us to see the function of each manager.
The viewport is now showing a number of objects which are represented by their names in the Object manager. Next to each object name is the object icon and a few other buttons and icons, the functions of which will become clear while working through this tutorial manual.

The Material manager contains all the materials used for the objects in this scene. Materials are used to define surface parameters like colour, reflectivity, transparency and many more.

- **To see what the materials really look like, select Render > Render View from the main menu.**

The Attributes manager lists all the properties of the currently selected elements. All parameters in here can be adjusted interactively. Changing the size of an object, for example, will be immediately visible in the viewport.

Using the icon palette to the left of the interface we choose what we want to modify (e.g. the object itself or perhaps just the object’s axis). The icon palette at the top of the screen contains tools to move, scale and rotate elements as well as rendering controls and the object creation icons. The icon palette directly underneath the viewport is used for animating objects.
The Status bar at the very bottom of the interface provides useful information about the icons or menu commands we are choosing, or about calculations that are being performed by CINEMA 4D.

**Summary**

We should now have a feel for the CINEMA 4D interface; the managers, the icon palettes and the viewports. We will get more practice in using these tools as we progress through the following sections.
Naming Things

Everything has a name and we’ll find it helpful to get into the habit of using meaningful names for our objects, materials and other bits and pieces within CINEMA 4D.

Whether we do it consciously or not, we all spend most of our day using names for things that are part of our everyday lives. How would we cope if we didn’t have names for the shops we shop in, the towns we live in or our children and the other people we know?

Sometimes, in context, it’s fine to say “Pass me that cup, please.” if there’s only one cup sitting on the table. However, if we want one from the cupboard in the kitchen we might have to say “Can I have the blue, china cup please?” or “Give Susan the Teletubbies mug.”

Names help us organize and identify things quickly and easily; they also help others know what we’re talking about.

Within CINEMA 4D many things have names, from objects to materials to tags, and it’s just as important to use meaningful names for these items wherever possible. So let’s take a look at how we name items in CINEMA 4D. We’ll start with objects.

► Select File > New to open a new scene.
Select Objects > Primitive > Sphere to create a sphere.

We can change the name of this object in a number of ways.

This sphere will appear in the Object manager with the name Sphere, and we should also see its name on the Basic page of the Attributes manager when Sphere is selected in the Object manager.

Select Sphere in the Object manager by clicking on its name.

In the Attributes manager, click on Basic to open the Basic Properties page.

Still in the Attributes manager, now double-click on the word Sphere to highlight the text, then type Ball and press the Return key.

We have changed the sphere’s name to Ball.

In the Object manager, double-click on the word Ball.

A dialog will open with the original name selected, ready for us to type in the new name. We can also access this dialog by selecting Rename Object from the Objects menu in the Object manager.
In the Name dialog, type in Eye and confirm the dialog by clicking OK.

Which method we use depends on the circumstances; if an object is already selected it can be easier to change its name in the Attributes manager. However, if we are browsing through the Object manager, a double-click on the object’s name will be a faster way to rename it.

So that’s how we name objects. Materials are named in a similar way. If we load materials from our hard drive or CD we should find that they are already named, but if we create one ourselves we may well want to give it a meaningful name.

In the Material manager, select File > New Material to create a new material.

By default, this material is named New; we can rename it in a number of ways.

In the Material manager, double-click on the name New below the material’s icon.

A dialog will open with the text New already selected, ready for us to type in a new name.

Type Pupil and then click on OK to confirm the dialog.

We have changed the material’s name to Pupil.
With the Pupil material selected in the Material manager, click on the Basic page in the Attributes manager.

Now we can rename the material by selecting the word Pupil and typing in a new name. Try it.

The final way of naming a material is by double-clicking its icon (as opposed to its name) in the Material manager; this will open the Material dialog which is a mirror of the information that’s available in the Attributes manager. Here we can type in a new name for the material and close the dialog.

The last things we’re going to learn how to name are tags. Tags are assigned to objects to add power and flexibility. We can name tags so that we know what they do. Here’s an example.

► In the Object manager, select the Eye object.
► Select New Tag > Display tag from the Object manager’s File menu.

This adds a display tag to the Eye object. This tag enables us to choose how the Eye object will be shown in the viewport. Let’s change the Eye’s display to make it a wireframe.

► With the Eye object selected, click on Tag in the Attributes manager.
Ensure that Use Display Mode is enabled and select Wireframe from the drop-down menu.

We’ve changed the Eye’s display to wireframe but the tag that has done this is still called Display; it might be useful if the tag’s name described what it is doing.

To be honest, we won’t want to name display tags normally, but there are other tags, especially selection tags, which it is very useful to name. The use of selection tags will not be covered here, but the method of naming them is the same as for display tags.

In the Attributes manager, click on Basic to open the tag’s Basic Properties.

Change its name from Display to Wireframe.

Now, when we click on this tag in the Object manager (to the right of the Eye object) we will see what it is doing by its name in the Attributes manager.

Summary

We’ve now seen how to name many items within CINEMA 4D and we should now have a good grasp of how to do this throughout the interface. Other places where it makes sense to name things is within the Timeline (double-click the name or select Objects > Rename Object), within the F-Curve window (double-click the name) and, most obviously, when we are saving our scenes.
Navigating in the Viewports

*CINEMA 4D gives us many different views on our work and various ways of changing those views.*

When modeling, we need to be able to see our object from more than just one side. In this part of the Quickstart tutorial we will demonstrate the various quick methods CINEMA 4D provides for us to move around in our scene.

- **Choose File > New** from the main menu to create a new scene.
- **Create a Figure** by holding down the mouse button on the Primitives icon (showing the blue cube) in the top toolbar and selecting the Figure icon from the command group.

Note the four little icons in the top right corner of the viewport. These icons are the quick navigation icons.

- In the viewport, hold down the mouse button on the first of these icons then drag the mouse around.

The first quick navigation icon enables us to move around in the view. The keyboard short-cut for this icon is the ‘1’ key (hold down this key and drag the mouse).

- **Hold down the mouse button on the second quick navigation icon, then drag the mouse again.**

This icon is used for zooming in and out. The keyboard short-cut for this icon is the ‘2’ key.
Hold down the mouse button on the third icon. Again, drag the mouse.

The third quick navigation icon lets us rotate around the active object, the figure in this case. The keyboard short-cut for this icon is the ‘3’ key.

Click on the last of the four quick navigation icons.

This will switch the current single view to the 4-view mode. This last quick navigation icon is different from the others — it is the Toggle View button, it enables us to switch from single view to 4-view mode and back.

In any of the viewports, click on the Toggle View icon to maximize that view.

We can also use the Page Up and Page Down keys on the keyboard to switch between single and 4-view mode. Note that when toggling to the single view, CINEMA 4D always chooses the view which is currently active. A view is active if it has a thin blue border around it. To make a view active, click into an empty area inside the view or click on its title bar.

Summary

We’ve seen how we can look at our scene from different angles and perspectives and how we can then move around within those views to center on the subject we’re currently working on. We may find that different arrangements of views will suit different tasks, from architecture to character work; CINEMA 4D adapts to our needs.
Creating and Modifying Objects

CINEMA 4D contains a large number of pre-defined objects or ‘primitives’ for us to work with.

This part of the Quickstart tutorial explains how to create and modify a simple cylinder using the Move, Scale and Rotate tools. We will also learn how to use the Attributes manager to adjust object parameters.

► Choose File > New from the main menu to create a new empty scene.

► In the top toolbar, hold down the mouse button on the blue Cube icon.

Holding down the mouse button on this icon opens the primitive objects command group where all primitive objects like cube, sphere, cylinder, torus, landscape and a few more are listed.

► Move the mouse pointer to the Cylinder icon inside the command group and let go of the mouse button.

A cylinder will be created in the middle of the viewport.

To create this cylinder we held down the mouse button and chose it from the command group. However, if we wanted to create a cube we could just click on the Cube icon without waiting for the command group to appear.
Apart from the cylinder, we can also see its red bounding box (which tells us that the
cylinder is currently selected), two orange handles and three arrows pointing outwards
from the middle of the cylinder. The arrows are the object’s X, Y and Z axes (red = X,
green = Y, blue = Z). The orange handles represent certain parameters of an object, in
this case height and radius.

► **Click and hold down the mouse button on the outer orange handle, then drag
to change the radius of the cylinder.**

Many objects in CINEMA 4D have orange handles. These handles let us interactively
change certain parameters of objects. In this case, we have changed the cylinder’s radius.
The orange handle in the middle of the cylinder would change its height. We can also see
these parameter changes in the Attributes manager.

► **In the Attributes manager, enable the Fillet option.**

The Fillet option gives the cylinder a rounded edge at the top and bottom. It also creates
a new orange handle near the top of the cylinder which can be used in the viewport to
control the radius of the round edge. All of the objects in the Primitives command group
(or in the the Objects > Primitive menu) can be modified in similar ways. Different objects
will have different parameters, many of which can be adjusted with the orange handles.
Click and hold down the mouse button on the arrow of the red axis, then drag the mouse left and right to move the cylinder along its X axis.

Clicking and dragging the red axis moves the cylinder along X.

By clicking on an axis, we can move an object along that specific axis.

Let go of the red arrow, then click and drag the mouse pointer around in an empty area of the viewport.

By clicking on an axis, we can move an object along that specific axis.

Let go of the red arrow, then click and drag the mouse pointer around in an empty area of the viewport.

By dragging in an empty area of the viewport (i.e. not selecting any of the axes) we can move the cylinder in any direction. The movement is not limited to one axis only.

Click on the Scale tool in the top toolbar.

The Scale tool is next to the Move tool in the top toolbar.

In the viewport, click in an empty space and drag the mouse left and right to scale the cylinder.

The Scale tool is used to scale any selected object or element.

Click on the Rotate tool in the top toolbar.

The Rotate tool is used to rotate objects or elements.

This tool enables us to rotate the cylinder around its origin. The yellow circle around the cylinder divides the viewport into two regions. If we drag inside the yellow circle, the cylinder is rotated around all its axes at once, depending on which direction we move the mouse. If we drag outside the yellow circle, the cylinder is rotated around an (invisible) axis which is perpendicular to the current view plane.
Summary

We’ve learnt how to take one of the pre-defined objects and add a fillet to it in order to create a rounded cylinder. We then saw how to scale and rotate this object. CINEMA 4D contains a wide range of pre-defined objects which may well be a suitable starting point for our 3D work, especially given the many editing tools available to us. In a later tutorial we will learn how we can treat these objects as a piece of clay that can be pulled and pushed into all sorts of interesting shapes.
Creating and Applying a Material

Once we have created our objects we need to bring them to life by adding texture to them.

New objects are always grey by default. To change this we need to create a material in the Material manager, adjust its colour, then apply it to our object. This section of the Quickstart tutorial shows how a material is created and applied.

► Choose File > New from the main menu to create a new scene.

► Create a Torus by holding down the mouse button on the Primitives icon (showing the blue cube) in the top toolbar and selecting the Torus icon in the command group.

► In the Material manager, select File > New Material.

The new material is just as grey as the torus at the moment.

► In the Material manager, double click the material preview to open the Material Editor.

In the Material Editor we can define all the aspects of a material. For now we want to change only the surface colour.
In the Material Editor, use the Color RGB sliders to change the colour to a bright blue. Set the R slider to 20%, G to 50%, B to 100% and additionally the Brightness slider to 100%.

![Use the colour sliders to mix a bright blue.](image)

The top three sliders define the colour value (R=Red, G=Green, B=Blue) and the Brightness slider is used to control the overall brightness of the colour.

If we click on the Color box to the left of the sliders we can use the operating system’s colour chooser to change the colour.

- **Close the Material Editor by clicking on its Close icon in the title bar.**

- **In the Material manager, hold down the mouse button on the blue material preview. Now drag it over to the Torus object in the viewport and release the mouse button.**

![The torus now has the blue material assigned.](image)

The torus now has the blue material assigned. The torus is now blue. If we have more than one object in our scene we may find it easier to drag and drop the material onto the object name in the Object manager instead.

**Summary**

We've added a very simple material to our object to understand the principle of using the Material manager. We'll learn about more complex materials, including using texture images, in a later tutorial.
Resizing Windows

There are many ways of customizing the CINEMA 4D interface; we will start with one of the simplest.

The CINEMA 4D interface contains many windows that butt up to one another. We may well want to adjust the height and the width of these windows to obtain a display that suits our needs. This will help our workflow.

- **To make the viewport smaller and the Material manager bigger, move the mouse pointer over the border between both windows until the mouse pointer changes to a resizing arrow.**

  Resize the view window to be able to see more materials in the Material manager.

- **Drag the window border upwards.**

  Resizing the Material manager can be useful when we are working on a complex scene which has a lot of materials. If we need more space in the view window, we simply drag the border back to where it was before.
Summary

Different jobs are made easier with different arrangements of windows; animation work will require us to focus on the Timeline while we may want to have a large Material manager window when texturing. We have seen that resizing windows in CINEMA 4D is very easy; later we will learn how to configure the views in other ways and how to save these arrangements as a layout.
Creating a new Icon Palette

Pictures are a great way of telling a story; CINEMA 4D enables us to use pictures in the form of icons.

Icon palettes consist of a number of icons collected together in a logical way, often for a particular purpose. CINEMA 4D comes with some pre-defined, ready-to-use icon palettes which may be added to and/or by extra modules or plugins or by our own fair hand. Here we will see how we can create a completely new icon palette.

- **Select Window > Layout > New Icon Palette from the main menu.**

A new icon palette has been created.

A new icon palette is created. It is empty at the moment.

- **Right click (command-click on Mac) on the empty palette and select Edit Palettes from the context menu.**

Choose Edit Palettes from the context menu to open the Command manager.

The Command manager will open and the empty palette (and also all other icon palettes) can now be filled with icons from the Command manager.
In the Command manager, click on the pulldown menu at top left and set it to Editor - Selection.

Set the pulldown menu to Editor - Selection.

By default the Command manager displays all the commands in CINEMA 4D. As there are so many of them, we can specify which group of commands we would like to see. We want to create an icon palette specifically for selection, so we choose to display only the selection commands.

Scroll down the command list until the Select All command appears.

Hold down the mouse button on the Select All command and drag it over to the empty palette. As soon as a small black bar appears in the palette and the mouse pointer turns into a little square, let go of the mouse button and the icon will drop into place.

Drag and drop the Select All command into the empty palette.
» Scroll up the command list and repeat the previous step for the Deselect All command.

Two icons have been added to the palette.

![Image of command list with icons added]

» We can now disable the palette editing mode by disabling the Edit Palettes option in the Command manager.

Disable the palette editing mode by disabling the Edit Palettes option.

![Image of command manager with Edit Palettes option deselected]

» Grab the palette by holding down the mouse button on the two small bars to the left of the icons, then drag it to near the bottom of the viewport. As soon as a black bar appears at the bottom of the viewport and the mouse pointer turns into a little square, let go of the palette and it will drop into place.

» Right click (command-click on Mac) on the palette and select Text from the context menu.

The Text option displays the command names next to the icons.

» Right click (command-click on Mac) on the palette and unselect Icons from the context menu.

The icon palette is now a text only palette for quick selection and delection.

![Image of text-only icon palette]

This disables the display of the icons so that only the text remains. Now we have a handy selection palette below the viewport which can save us a lot of time when working with lots of objects because we no longer have to use the menu to select and deselect objects or other elements in our scene.

**Summary**

Icon palettes are another great way of improving our CINEMA 4D workflow. We’ve seen how easy it is to create our own palette so that we can organize the display in a way that suits us. It’s just as easy to edit the existing palettes, making CINEMA 4D a tool that works for us, not against us.
Integrating the Material Editor with the Layout

The ability to dock and undock dialogs adds to the flexibility of the CINEMA 4D interface.

Depending on the nature of the project on which we’re working, we may want quick and easy access to some of CINEMA 4D’s dialogs. For instance, when texturing it can be useful to have the Material Editor always available on a tab, rather than having to double-click on the relevant material icon. Docking to the rescue.

- **In the Material manager, select File > New Material.**
- **Double click the new material preview to open the Material Editor.**

We want to integrate this editor window into the layout so that we no longer have to open and close it when editing materials.
In the Material Editor, hold down the mouse button on the little blue pin (the drawing pin icon).

The pin icon enables us to dock and undock windows.

>> Drag the drawing pin icon with the mouse until it is to the right of the Attributes tab and let go of the mouse button.

As soon as we let go of the mouse button, the Material Editor will be inserted between the two tabs. Now it has become *docked* and is no longer a floating dialog.

We can now switch back and forth between the other tabs and the Material tab, so when working with materials we no longer need to open and close the Material Editor. It is now well integrated with the layout. If another manager is in the foreground and the Material Editor is hidden, just double click any material preview in the Material manager and the Material Editor will pop to the front.

**Summary**

We have seen how easy it is to dock dialogs within CINEMA 4D, using the drawing pin icon. If we wish to undock a dialog, we simply click over the relevant drawing pin icon; a menu will appear from which we can select Undock. We can dock and undock dialogs to suit our way of working and the task in hand.
Saving and Loading Layouts

Having designed the interface the way we want it, it would be rather frustrating if we couldn’t save it.

The ability to load pre-designed layouts into CINEMA 4D is a vital tool in our armory, enabling us to quickly adjust the workflow for different projects. We might have a layout for architectural work, for visualization, for character animation, for texturing and painting. CINEMA 4D enables us to save and load layouts with ease; we can even adopt our new layout as the default one.

- **To save a layout as the new default layout, select Window > Layout > Save as Default Layout.**

  The original, default, layout can be restored by selecting Window > Layout > Reset Layout.

- **To save a layout as additional to the default layout, select Window > Layout > Save Layout as... and type in a name for this layout.**

  Ensure that the layout is saved into the Prefs folder in the CINEMA 4D main folder. If we save it anywhere else, this layout won’t appear in the Layout drop-down list.
Next time we start CINEMA 4D and we want to recall a layout, if we click the icon in the top left corner of the CINEMA 4D interface (the layout switch icon) we can select it from the list. It will be displayed as the filename we gave it earlier.

Alternatively we can go to the Window > Layout menu and select it from the bottom of this menu. To change an existing layout (e.g. the Animation layout), load it from the list, adjust it and then save it in the Prefs folder using the same filename.

Summary
The best way to learn about layouts is simply to experiment. We should not be afraid to try out new ideas to see how they fit our workflow; at any time we can reload a previously-saved layout or revert to the default layout.
Metaballs

Metaballs provide us with a modeling method which is more akin to working with clay than the traditional 3D modeling method of building objects from primitives and polygons.

Rather than working with exact dimensions and precise tools, meta-modeling enables us to create organic looking models by lumping together virtual clay. The most common type of meta-modeling objects are metaballs. Each sphere you add to a metaball will increase the mass in the location of the sphere. By placing several spheres in a rough shape, the metaball will take these and merge them to create a smooth outer skin.

The other type of meta-modeling available makes use of spline objects. In the case of splines, the whole length of the spline is turned into a sausage shaped lump of clay which can be manipulated and combined with metaballs. In this tutorial we will be creating a gummy bear using a combination of metaballs and metasplines.

To keep our scene well organised, as we work through this tutorial we should always give our objects a reasonable name. For example, when we create a sphere for the left arm, we might name it something such as Left Arm. Throughout this chapter we will leave the naming of objects completely up to you. Further guidance on naming objects can be found in the relevant section of the Quickstart chapter earlier in this tutorial manual.
Click and hold down the mouse button on the Modeling Aids icon, then select the Metaball object from the command group and release the mouse button.

Metaballs make use of normal objects such as spheres and splines. These normal objects are turned into metaballs by placing them inside a Metaball object. The Metaball object stores all the settings, such as how much mass each object adds to the model and what level of detail should be used for the viewport and when rendering.

Click and hold down the mouse button on the Primitive Objects icon, then select the Sphere object from the command group and release the mouse button.

When an object is added to a metaball, every point on the object will add an equal amount of mass to the structure, with the exception of a sphere. Spheres are treated as a special case, instead they will only add a single point of mass to the object. The sphere’s radius is used to determine just how much mass is added.

In the Object manager, ensure that the Sphere object is selected, then select Edit > Copy followed by Edit > Paste.

In the viewport, drag the spheres so that they sit either side of the centre line, as illustrated below.
In the Object manager, drag both Sphere objects and drop them into the Metaball object.

CINEMA 4D’s hierarchical modeling allows many objects to be added to the Metaball.

Objects become part of the metaball only when they are a child of the metaball, otherwise they continue to act as normal objects independent from the metaball object. If the spheres are close enough we should see them begin to merge into a single, gloopy object. If we move the spheres around in the viewport we’ll get a feel for how metaball objects interact with one another. We will see that they do not need to be touching before they can stick together. To move the spheres back to their original positions we can select Undo from the Edit menu.

In the Object manager select the Metaball object, then in the Attributes manager reduce the Editor Subdivision level to 20.

The Attributes manager enables you to alter various settings quickly and easily.

The Editor Subdivision level decides the distance between points and polygons and how often they should be added to the object within the 3D world. The lower this number, the more detail will be present. Think of it as a huge 3D grid which the metaballs are filling in — the lower the Editor Subdivision level, the more dense the objects will appear.

Metaballs require lots of processing power to calculate, hence the reason for being able to specify a lower detail level in the viewport. Depending on the speed of your computer, you might want to use a higher Editor Subdivision level. Typically keep the level between 10 and 40 when working with the default scale.
Add two more spheres from the Primitive Objects command group and reposition them as shown in the diagram.

In the Object manager, drag both new spheres and drop them into the Metaball object.

The Metaball object can hold a virtually unlimited number of spheres.

These two new spheres will form the gummy bear’s arms and hands; the previous two already inside the Metaball object will be its legs and feet. Only a single sphere is needed for each limb because the moulding process of jelly sweets removes any intricate details. If you haven’t already done so, now would be a good time to give these objects appropriate names (double click on their names in the Object manager).
Add a fifth sphere into the scene. In the Attributes manager, increase its radius to 200 then drag it into the Metaball object to join the other four spheres.

This new sphere will become the big chewy bit of our juicy sweet.

In the Object manager, select the Metaball object and in the Attributes manager increase the Hull Value to 180.

By now you may have realised most changes can be made in the Attributes manager.

The Hull Value tells the metaball how much volume each sphere should add to the whole object. If this percentage is too low then there will be very little definition in the model. This value needs to be increased to a point where the mesh sticks closer to the original spheres.
► Add another sphere to the scene and position it above the largest sphere.

Another sphere makes our bear’s head.

► In the Attributes manager, increase the radius of this new sphere to 150.

► In the Object manager, drag this sphere into the Metaball object

The list of spheres that make up the bear is beginning to grow.

Just like the other objects in the scene, this new sphere should merge into the rest of the metaball structure, this time becoming the head of the gummy bear.
In the Object manager, select both the left and the right arm spheres (hold down the Shift key while clicking to multi-select) then duplicate them by selecting Copy then Paste from the Edit menu.

Copy and Paste commands can speed up your workflow.

In the viewport, move the two duplicated spheres upwards.

Ensure the Model tool is selected.

There is an importance difference between the Model tool and the Object tool. To learn more about this difference, please see the Animation section of the Tips and Tricks chapter at the end of this manual.

Select the Scale tool.

Scale the spheres, then position them so that they appear similar to those in the illustration below.

The quickest way to make the ears is to duplicate similar existing parts.
In the Object manager, drag the new spheres and drop them into the Metaball object.

These two new spheres will be the bear’s ears.

In the Object manager, select both ear spheres then copy and paste them.

In the viewport move both of these new spheres forwards and then use the Attributes manager to scale the radii of these spheres to half of their original sizes.

Spheres can subtract from, as well as add to the object’s size.

In the Object manager, ensure that both new spheres are selected then select File > New Tag > Metaball Tag.

Tags can be applied to individual objects or groups to add extra parameters for more control.

A metaball tag can be used to give certain objects different properties to the others. Metaball tags allow us to specify how much volume a particular object should add to the rest of the Metaball mesh — or, indeed, should subtract from the mesh.

In the Object manager, drag both spheres into the Metaball object.
In the Object manager, select the metaball tags for both new spheres (shift-click) and in the Attributes manager enable the Negative Influence option.

Negative Influence can be used to make certain spheres repel the metaball surface rather than add to it. This is useful for creating dips, dents and grooves in an otherwise convex curved surface. Here, the two smaller spheres are being used to create a more hollow forward facing ear.

In the Object manager, select both of the spheres that have metaball tags and duplicate them by using Copy and Paste from the Edit menu.

In the viewport, move each sphere down and forward so that they become the bear's eyes.

In the Object manager, drag these two new spheres into the Metaball object. Just as with the ears, two spheres with a negative influence are going to be used to create dents in the jelly mesh. This time they are going to create eye sockets.
Add an Arc spline from the Spline Presets command group.

There are several ready-made splines; the Arc spline creates a 90° curve.

In the Object manager, drag the Arc spline into the Metaball object.

Virtually any object can be added to a metaball.

In the viewport, rotate and position the Arc spline as illustrated below. Use the Attributes manager to scale it to the required size.

Smile!

The mouth of our gummy bear is created by adding a spline object into the metaball. When a spline object is added to a metaball, it adds mass all along the length of it and creates a sausage shape.
In the Object manager, copy a metaball tag to the Arc spline by holding down the Control or Ctrl key on your keyboard and dragging one of the existing ones and dropping it onto the Arc spline.

Most objects and elements within CINEMA 4D can be duplicated by holding down the Control or Ctrl key and then dragging that item to another place. The original will be left untouched and a duplicate will appear wherever you drop it.

In the Object manager, select the Arc spline’s metaball tag and in the Attributes manager set its Strength to 50% and Radius to 1.

A slight indentation is needed for the mouth area so the Strength is reduced from 100% down to 50%. When using splines, each and every point along the spline is treated as a sphere, so a typical spline could effectively add the volume of dozens of spheres to an object. The radii of these mini-spheres can be reduced by decreasing the Radius value in the metaball tag.
In the Object manager, copy and paste one of the eyeball spheres then move it to the front and centre of the bear’s body.

Cute gummy bears need cute belly buttons.

The finishing touch for our Gummy Bear is a cute belly button. This is made by taking one of the eyeball spheres, which already has a metaball tag with negative influence set, duplicating it and moving it to the appropriate position.

In the Object manager, select the Metaball object and in the Attributes manager set the Render Subdivision level to 3.

A lower subdivision gives a higher level of detail. In this case we are changing the level of detail which is used when the object is rendering.

In the 3D viewport menu, Select Display > Level of Detail > Use Render LOD for Editor Rendering.

By default, CINEMA 4D will use the editor LOD (level of detail) when you render an image into the viewport (i.e. the editor). You can override this and force it to use the render level of detail.
To see the model in a higher level of detail, and without slowing down the interface, we can quickly render the object in the 3D viewport. To make the render a higher quality we first had to select a low subdivision for the Metaball object and secondly tell the editor (viewport) renderer that it should use the final render quality instead of the viewport settings.

**Summary**

The modeling of the gummy bear is now complete. As this is a modeling tutorial, this is where we will end. It may appear fairly bland, but rest assured that you have created a very high quality object mesh which can easily be textured and animated using the rest of CINEMA 4D's tools.
HyperNURBS 1

Creating a very simple, low resolution model which has smooth curves and crisp corners is a fantastic way to model. Learn to be efficient with HyperNURBS.

NURBS modeling is a method of creating objects using a relatively low number of control points to influence the direction of the mesh. HyperNURBS provide us with the ability to make smooth and intricate models without having to deal with a complex object. To understand exactly what is happening, let’s take a look at a regular 2D spline object.

This is a B Spline, the points influence the direction.

If we look at the diagram above, it should illustrate exactly what is happening to the surface of the object. There are just five control points to describe the direction of the spline. Ignoring the first (starting) and last (ending) points, all intermediate points will pull equally on the spline as if they have a magnetic influence. Unless a point is added directly into the existing path, the actual spline surface will very rarely pass through one of the control points.
HyperNURBS take B Splines one step further by adding a further dimension to the object. Instead of using the points to just influence a line which has no volume, HyperNURBS are capable of adding real geometry to the object. Let’s take a very simple example of how this works.

- **Add a Cube object to the scene by clicking the Cube icon in the top toolbar.**

For our first small example a cube will be used to control the HyperNURBS object. Note that a cube object has eight points, one at each corner.

- **Add a HyperNURBS object to the scene from the top toolbar.**

The HyperNURBS object has no actual shape, instead it is used to affect other models. Once the Cube and HyperNURBS objects have been added to the scene they appear in the Object manager. The Cube should be on top as this was the first object we added.

- **In the Object manager, drag the Cube object and drop it into the HyperNURBS object.**

Once we have dropped the Cube inside the HyperNURBS, in the viewport we will see it instantly change into a sphere. The reason why is because now that the cube is inside the HyperNURBS, the points on the cube are being used only to pull on the surface of the object. Each point of the cube is pulling equally on the surface of the mesh, because all of the forces are equal, and this creates a sphere inside the cube.

- **In the Object manager, ensure the Cube object is selected and then click the Make Editable button in the left-hand toolbar.**

Once an object has been made editable we will be able to permanently change its shape by moving points, polygons and edges. The reason for not making an object editable to start with is so that we can more easily adjust its values, such as how many divisions it has or how smooth the edges should be. These changes cannot be made after making an object editable.
Select points mode by clicking the Points button in the left-hand toolbar.

We can edit three elements of an object: points, edges and polygons. By editing just one part of the object we can deform the model into different shapes.

In the viewport, select one of the points (by clicking on it once) then drag the point to change the shape of the sphere.

As we pull the point away from the sphere, notice that the sphere becomes more influenced in that direction. This works just as well if we push the point inwards to make an indentation.

With the point still selected, hold down the period (fullstop) key then drag the mouse button to the right.

By holding down the period key we are altering how much influence that particular point has over the final object. This is known as weighting. The more weight we apply to a point, the more the rest of the object will be attracted towards it.
Summary

We have seen that by using HyperNURBS we can quickly create smooth models from objects with only a few points. From here we should experiment for a while by moving around the points and polygons of the cube to get a feel for how they influence the shape. We should also try pushing points into the object so that it becomes concave. We will see that by using just these eight points we can make all manner of interesting shapes.
HyperNURBS 2

Realistic objects are easily made with the wide range of modeling tools available to us in CINEMA 4D. Here we’ll see how to add the finishing touches to an existing model.

We have covered the basics of how HyperNURBS work and now it is time to use this knowledge to create a practical object. We are going to create a standard American 2-pronged plug. The reason why we have chosen this is because it enables us to make use of the Bridge tool in an interesting way.

► Load the scene called ‘Plug Before.c4d’ from the Modeling folder in the CINEMA 4D Tutorials folder.

Rather than making everything from scratch, we have pre-modelled the main part of the plug. From here we will need to use various tools to complete the model.

► Select polygon mode by clicking the Polygon icon in the left-hand toolbar.

The polygon mode opens up lots more new tools for us to use. Most of these tools can be found in the Structure menu.
In the viewport, hold down the Shift key then select the two centre polygons, as illustrated below.

We must select the polygons on which we wish to perform a function.

By holding down the Shift key and clicking, we are able to select multiple items. These items may be points, polygons, edges or entire objects. To perform an operation we must first select which areas we want to affect. In this case we have chosen the base of the two prongs.

From the Structure menu select Extrude, then in the viewport drag to the right until two prongs have extended.

Extruding polygons will either bring them out or push them in.

The Extrude command will take an existing polygon or edge and pull it outwards by creating new polygons wherever needed. This is used to create extra detail without disturbing the surrounding area. When we drag the mouse, moving right will extrude outwards while moving left would push inwards; these directions will work regardless of the orientation of the model.
In the viewport, drag once more to add an extra section to the already extruded prong.

A further extrusion will add more detail to the end.

This extra extrusion is in preparation for the next few steps and will enable us to add more detail to the end of the prongs. Note that it is important that we avoid making triangles when using HyperNURBS as they may cause rendering artifacts.

- Select either the Move, Scale or Rotate tool then select the four square polygons we have just made (shift-click to multi-select).

Select the four polygons by shift-clicking on them.

We may only click to select an item if we are in one of the basic modes, such as move, scale or rotate. While we could use one of the selection tools, in this case simply clicking on the polygons is the easiest method.
Press the ‘i’ key then drag the mouse left to pull the polygons inwards slightly.

An Inner Extrude has been used.

By pressing ‘i’ we have used one of the many keyboard short-cuts to enable a different mode. By default, pressing ‘i’ will enable the Inner Extrude command. When we drag, we can either move the mouse to the right to enlarge the new polygons or left to shrink them.

Press the ‘b’ key then drag a straight line between the corresponding corners of the two selected polygons. Do this for each prong.

One of the lesser known functions of the Bridge tool is its ability to create a tunnel through an object. The Bridge tool links one section to another. If no polygons are currently there, then new ones will be made. On the other hand, if you link one point to another and a polygon already exists there, a tunnel will be made from one side to another. We need to select only the correct corresponding corners so that the Bridge tool knows which parts it should link.
Select the Edge tool in the left-hand toolbar.

Edge selection mode enables us to pick the edges of models.

Select the Live Selection tool in the top toolbar.

The Live Selection tool enables us to paint a selection.

In the viewport, with the Shift key held down select the eight surrounding edges of the holes we made on both prongs.

Select the surrounding edges to be weighted.

When weighting edges we must first select which ones we want to alter. We then use either the Live Selection tool or the Weighting tool to choose the edge’s influence.

In the Active Tool manager, ensure the Strength slider is set to 100% then click the Set button.

We may need to scroll down in the Active Tool manager to see the Weighting section.
Set the weight of the edges to 100% and the prong holes now have a sharp edge.

Rather than holding the period key to set the weight of the edges, we can use the Live Selection tool to accurately select a weight for the edges. This is useful if we want extreme levels of weighting.

From this point of the tutorial we may need to disable the HyperNURBS in order to see certain points or edges. An example of this is at the base of the prong where it meets the rest of the plastic plug. Because of its concave nature, this area has a large curved surface covering it.

- To turn off the HyperNURBS, in the Object manager click its green check mark (to the right of the HyperNURBS object name) so that it becomes a red cross.
- In the viewport, select the edges that are to be weighted.
- To turn the HyperNURBS back on, in the Object manager click its red cross so that it becomes a green checkmark.

- Using the Live Selection tool, select the four large polygons around the base of each prong, plus the smaller top and bottom edges.
In the Active Tool manager, ensure the Strength slider is set to 100% then click the Set button.

Remember that we can set exact weights by using the Live Selection tool and the Active Tool manager.

In the viewport, select every other rib and the large surface on the top and bottom. Refer to the illustration below to see where these parts are.

These will form the grip and the flexible ribs.

Most electrical plugs have an indentation of some sort to help with gripping it. Also often found are some bendy plastic ribs which lead onto the electrical wire, these are to prevent the wire from being pulled out of the plug.

In the viewport, click the right mouse button and select Extrude from the context menu.

There is more than one way to access the same command. Not only can we use the menus, toolbars and short-cuts, but all objects will also have a context sensitive pop-up menu which contains the most used commands for that particular object.
In the viewport, drag the mouse left until a set of ribs are made near the cable.

The Extrude tool has created the ribs and a dent near the prongs.

The exact sizes and proportions of these extrusions are unimportant in this case. We should use our judgement to imagine what the plug should look like.

Select the Move tool then, in the viewport, select the ribbed faces that point outwards, ignoring the vertical ones.

Selecting the faces prior to performing edge weighting.

Select Convert Selection from the Selection menu and select Polygons to Edges.

When we weight a polygon, we are effectively weighting the surrounding edges and points. If the polygons themselves were weighted then they would still have sharp corners due to the weight of the points. By converting the polygon selection to an edge selection, we can safely weight the edges without the points being weighted with them.
Select the Edge tool in the left-hand toolbar.

Hold down the period (fullstop) key then, in the viewport, drag right until the ribs become squarer, but not with sharp edges.

The now weighted ribs have more definition than before.

We want the ribs around the cable to have more defined edges, but we don’t want them to have sharp angles on them, which would look unnatural.

Select polygon mode by clicking the Polygon icon in the left-hand toolbar.

In the viewport, select the four inner polygons which were created on the large area on both sides.

Remember that we may need to disable the HyperNURBS to see the parts we are trying to reach.

The large gripping area needs a bit more shape.
Hold down the period key then, in the viewport, drag to the right until the corners are less rounded.

This shape is typical of a cheap moulding process.

When plugs are moulded, there will rarely be a perfectly sharp corner. Here we have made it almost a sharp corner, but left a slight bevel on the edge.

Switch to edge selection mode (select the Edge tool), then select the four edges at either end of the cable.

These edges will have weighting applied.
Hold down the period key and drag all the way to the right to fully weight these edges.

Now that the wire has a constant profile along its complete length, our plug model is complete.

Summary
Here we have learnt several important ways of working while modeling with HyperNURBS, the most important thing being that we try to use quadrangles wherever possible. Triangles can be used where essential, but they may cause shading problems if we start using them in large quantities.

If we are able to keep a clean mesh then we will find that we can create smooth, flowing models with very few polygons. These low resolution models will help us immensely later when we begin animating them — the simple fact is that a 500-polygon model is easier to handle than a 5000-polygon one.

We should also note that not only can we weight polygons and edges, but also points. Although we didn’t weight any points with this model, we can gain even more control by weighting the individual points of an object.

If you have purchased the MOCCA character animation module, or a bundle which contains it, then you will find in the MOCCA manual further tips for modeling with HyperNURBS which prepare you for animating them.
NURBS

By using spline vectors we can create and describe the profile of an object. CINEMA 4D can then fill in the gaps to create our model.

NURBS (as opposed to HyperNURBS) work by applying geometry to one or more splines. A Sweep NURBS, for example, will take the first spline object and treat it as a profile (i.e. a cross-section) then it will take a second spline and stretch a skin along its length to create a long, flowing object. Think of a water flume at a swimming pool, or a tail flowing behind a kite. A third spline can optionally be used to define the thickness and angle of each cross-section as it flows along the path.

In this tutorial we will be using the Loft NURBS. This kind of NURBS enables us to take an unlimited number of spline profiles and ‘skin’ them together. A typical example of this might be a ship hull, which is what we will be creating here. Hulls are an ideal candidate for Loft NURBS as they change size, shape and position along the entire length of the vessel and are often smooth and flowing shapes to allow streamlined movement.

► Add a Loft NURBS object by clicking on the NURBS tool then releasing the button when the pointer is above the Loft NURBS icon in the command group.

The Loft NURBS object is a container which modifies its contents to create a new model.

The Loft NURBS object is what we will use to tell CINEMA 4D that it should be stitching all of the cross-sections together. Without the Loft NURBS, any cross-sections we make will remain as individual splines.
Click and hold the mouse button on the Spline Presets toolbar icon and add a circle spline from the command group.

If we click and hold the mouse button on a toolbar icon, a list of other similar types of objects will appear. If, however, we quickly click once on the toolbar icon, whichever icon is shown will be selected and added.

Click the Make Object Editable icon in the left-hand toolbar.

The preset splines are not editable to begin with. This tool will make them editable.

When the circle is in a parametric state it cannot be manually edited; for us to be able to change its shape, it must first be made editable. Note that after making an object editable we will no longer be able to change parametric details such as subdivision or radius.

Select points mode by clicking the Points icon in the left-hand toolbar.

The points of a spline can be edited only when CINEMA 4D is in point editing mode. Once this mode is enabled we should see our spline turn yellow and red to show the order of the points in the spline. The spline begins at the yellow end and finishes where it turns red.
Select the top point by clicking on it, then delete it.

After deleting the top point of the circle spline, the shape already begins to look like the profile of a ship’s hull.

We are creating a ship hull so none of the points along the top of the object will be needed as they would be covered by the rest of the ship, should we continue to model the entire vessel.

Ensure the Circle spline is selected in the Object manager, then in the Attributes manager disable the Close Spline option.

Closed splines would create a tube type object so we are disabling that option.

To open up the inside of the hull we need to tell CINEMA 4D that it shouldn’t seal the top of the spline. This will allow us to later fill in the interior with decks or cargo. By removing unnecessary surfaces from the top of the hull we are also reducing the system requirements for processing time and memory.
In the viewport, select the left-most point of the Circle spline by clicking on it, then select Set First Point from the Structure > Edit Spline menu.

The left-most point is where we want the spline to start from.

By default the open side of the circle spline is on the right-hand side rather than the top. The open section of a spline will always be between the first and last points, therefore we must tell the left hand point that it should be the first point.

Switch to the Front view by pressing F4 on the keyboard.

Either of the quad views can be made larger by pressing the keys F1 to F4.

The keys F1 to F4 enable us to quickly change between Perspective, Top, Side and Front views; pressing F5 will give us all four views simultaneously.
In the viewport, select the bottom point of the circle spline.

Ship hulls are not completely rounded, their shape often adds stability.

While holding down the Shift key, drag each of the pink tangent handles upwards one at a time.

Pink handles on a spline object enable us to fine tune the path that the spline takes through each point.

This pointed area at the bottom of the spline will become the keel of the hull. By holding down the Shift key we will be able to position each pink tangent independently; without the Shift key held down, when we move one of the tangent handles the other would move in the opposite direction.

Press F1 to switch to the 3D view, then select the Model tool in the left-hand toolbar.

Use the model tool to easily manipulate an entire object.
In the Object manager, drag the Circle spline and drop it into the Loft NURBS.

NURBS objects take a collection of splines and turn them into a useful model.

Loft NURBS work by skinning together any spline objects which are direct children. To make an object a child of the Loft NURBS we need to drag it into the object so that it becomes a child.

With the Circle Spline selected in the Object manager, select Copy then Paste from the Edit menu.

In the viewport, move the newly created spline backwards along the blue Z-axis (towards the camera) by 200 units.

Using several hull cross-sections you can create an entire ship.

In the Object manager, drag the new Circle spline into the Loft NURBS object and drop it above the original Circle spline.

The hull is skinned in the order of the splines.

Notice that the two spline objects have been joined with a sheet of ‘skin’. The rest of the ship hull can be created by repeating this process to extend the surface.

The order of the spline objects within the Loft NURBS is very important. The order in which they are displayed in the Object manager is the order in which the surface will flow. If you have several splines in the Loft NURBS out of sequence, the 3D mesh will flow incorrectly between them.
In the Object manager, paste two more copies of the cross-section and in the viewport move each one further along the Z-axis as shown in the diagram below.

The rest of the ship hull is created by placing multiple cross-sections.

In the Object manager, drag each spline section and drop it into the Loft NURBS object, starting with the one closest to the original.

The sections must be added to the Loft NURBS in the correct order to ensure a smooth flowing mesh.

When we have completed these two steps, we should end up with a much longer ship hull by having four identical spline objects inside the Loft NURBS.
In the Object manager, select the second spline in the Loft NURBS object and in the viewport increase its size by using the Scale tool until it is around 500 units wide in the X-axis.

A wider cross-section in the middle will give more room for cargo aboard this ship.

The centre part of the ship will be the largest so its cross-section needs to reflect this. By increasing this spline’s size the surface will bulge and stretch in order to accommodate it. The exact size of the middle cross-section is not important, so don’t worry too much about the exact values.

In the Object manager, select the top-most spline object and duplicate it by selecting Copy then Paste from the Edit menu.

Drag this new spline object into the Loft NURBS, ensuring it is dropped at the top of the list of splines.

In the viewport, scale it to a size of zero by using the Scale tool.

A zero sized spline will cause all connecting surfaces to come to a sharp point.

This new spline object will be used to seal off the bow of the ship. (That’s the front end to us land-lubbers.)
In the viewport, move the zero-size spline object further down the Z-axis until it creates a more pointed front end for the vessel.

By shrinking one of the spline segments to a size of zero, this will seal off one end of the hull by squeezing all of the polygons together. The further away this tiny spline is, the faster it will make your ship look.

In the Object manager, duplicate this zero-size spline by copying and pasting it, then drag it into the Loft NURBS and drop it at the bottom of the list.
In the viewport, move the selected spline along the Z-axis until it is at the rear of the hull.

Keep the back end short to differentiate between the front and back.

The other end of the ship hull also needs to be sealed. The rear of a ship hull is usually far more blunt than the front, so it has been placed almost level with the original spline to create a flatter aft.

In the Object manager, select the Phong tag for the Loft NURBS.

We're smoothing the model by applying a Phong tag.

The Phong tag enables flowing surfaces to appear much smoother than they actually are, without the need to add extra geometry.

In the Attributes manager enable Angle Limit and set the Phong Angle value to 40.

Limit the surfaces which will be smoothed with Phong shading by specifying a maximum angle.

By imposing a limit on the Phong tag we can instruct CINEMA 4D to smooth only those angles which are less than 40 degrees. Therefore anything that changes direction more sharply than this will have a well defined crease.
Summary

If all has gone to plan, we should have a completed ship hull. Don’t worry if it doesn’t look exactly like the one illustrated at the start of this tutorial, making it identical is not important. The important thing is that we have gained valuable experience with many of CINEMA 4D’s NURBS modeling tools.

The way in which we use splines to create an object is entirely down to us. We can either do it manually in the viewport, or we can make use of the Coordinates and Structure managers to enter exact dimensions. However accurate our model needs to be, we will find that CINEMA 4D will cater for us.

The other types of NURBS, such as Lathe NURBS, are also very easy to use and you should refer to the Reference Manual to learn about these other NURBS tools.
Basic Texturing
Sky

*Here we’ll learn how to use blurring and tiling techniques to create a background to encompass our scene.*

A well textured object can make or break an image and the texturing aspect of 3D work is often overlooked. Clever texturing, however, can save us a lot of modeling time. In this chapter we will learn two things. Firstly we will learn how to texture several objects in different ways using different techniques. And we will also learn that the other half of the texturing process, the more technical half, is placing textures onto our models in a sensible way. We will need to tell CINEMA 4D exactly how we want our textures to be placed on the surfaces of objects because the default positioning will rarely be suitable. A classic example of this is placing a flat label texture onto a certain place on a cylindrical bottle, something we will do right at the end of this Basic Texturing chapter.

To keep our scene well organized, we should always give our materials a reasonable name by typing it into the Name box in the top left corner of the Material Editor. When our scene begins to get larger we can easily lose our way though hundreds of similar materials, so we should try to get into the habit of naming everything while we still remember what it is. For example, we might name this material Sky or Background. Throughout this chapter we will leave the naming of materials completely up to you. Further guidance on naming materials can be found in the relevant section of the Quickstart chapter earlier in this tutorial manual.
Load the scene named ‘Party Before.c4d’ from the Texturing - Basic folder in the CINEMA 4D Tutorials folder.

This scene contains several untextured objects.

In this scene we will find all the objects required to complete this Basic Texturing chapter. First we will texture the sky (i.e. the background for the entire scene). For this we will use a suitable image and blur it to give the impression that the camera has not focussed on it.

In the Material manager, select File > New Material. To open the Material Editor, double click on the new material that appears.

Click the Image... button and use the file dialog to open the file named ‘party.tif’ from the tex folder in the Texturing - Basic folder.

The 2D image will be displayed on the right-hand side of the Material Editor and a 3D preview of it top left.

In the Texture pane, select SAT from the Sampling popup menu and set the O (Offset) value to 20%.

The image for the background is blurred using the Offset value.

Interpolation is used when the image we have chosen for our texture does not exactly match the resolution at which we wish to render. SAT is a type of interpolation which is used to render textures at a very high quality.

We are blurring the texture with the Offset value to give the impression that the camera is not focussed on this part of the image. The technical term for the blurring of objects in the background (and sometimes in the foreground) is called depth of field, or DOF for short. Although CINEMA 4D does have a special DOF function, it can be quicker and easier to blur a background texture manually.
Drag the material from the Material manager and drop it on the Sky object in the Object manager.

The image will now become visible in the viewport and the Attributes manager will display several parameters which define how the texture is projected onto the Sky object.

In the Attributes manager, ensure the Tag Properties page is displayed and set the Tiles X value to 2.

This repeats the background image once horizontally (i.e. 2 tiles), making it appear less stretched. We should be able to see the effect in the viewport instantly.

In the Attributes manager, Tag Properties page, enable the Seamless option.

Tiling a texture can result in visible seams where the texture edges meet one another. The Seamless option helps to avoid this by mirroring each texture tile instead of simply placing them next to one another.

Summary

We’ve learnt how to create a material and give it some texture by applying an image to it. We then saw how to add some effects to this material to achieve a sky effect and, finally, we learnt about tiling, to extend our material over a wide area, without joins.
Floor

For a floor material we need extra properties so that, for example, we can see the reflection of objects placed on the floor.

Many surfaces reflect light as well as absorbing it — it’s this that allows us to see our feet in the door of a stainless steel oven or a vase of flowers reflected in a glass table. Transparency is another property that is important when creating believable objects — many materials are inherently transparent. We may also use this effect to blend surfaces together. To create this blending we can use an Alpha channel, a material property that enables us to mask out parts of an image or scene.

For the floor we want to create a reflective material and also make the edge of the floor transparent so that it blends into the background with a soft falloff. To define this falloff we will use a texture loaded into the Alpha channel of the material we use for the Floor.

- If you haven’t already got it open, load the scene named ‘Party Before.c4d’ from the Texturing - Basic folder in the CINEMA 4D Tutorials folder.

- In the Material manager, select New Material from the File menu. To open the Material Editor, double click on the new material that appears.

- On the Color page, set the RGB sliders to a pale blue (R=80%, G=90%, B=100%, Brightness=60%).
Click the checkbox next to Reflection to enable reflections for this material and display the Reflection page. Set the RGB sliders to a very dark blue (R=85%, G=90%, B=100%, Brightness=20%).

This makes the floor object just a little bit reflective, like a shiny, plastic surface; anything that reflects in the floor will also become tinted slightly blue. Alternatively, we could leave the RGB sliders at 100% each and objects would then be reflected with their original colors. We could even have a blue surface reflect objects with a red tint by setting the RGB sliders on the Reflection page to red. Although in reality this would be difficult to achieve (if at all possible), this technique is great for creating all manner of visual effects and interesting surfaces.

Click the checkbox next to Alpha to enable the alpha channel for this material and display the Alpha page.

Using a material’s alpha channel we can fade out, or even completely cut out, parts of an object’s surface. These parts will then be rendered as slightly to completely transparent. (Note that this technique is not the same as making an object transparent using the Transparency channel.)

Click the Image... button and use the file dialog to open the file named ‘floor-alpha.jpg’ from the tex folder in the Texturing - Basic folder.

This image has a black circle with a soft edge on a white background and it will define which areas are opaque and which are transparent.

In the Texture pane, enable the Invert option.
We can use a texture to control the opacity of a material by either defining one color which is cut out like a stencil, or by using the strength of brightness (S) to make the surface more or less opaque. Disable the Soft option for the stencil method and click in the area in the square texture preview which you want to cut out. Alternatively, if the Soft option is enabled you can use an image which has its own alpha channel. This alpha channel is then displayed in the square preview instead of the actual image.

When using an image to define the alpha channel, normally a black area in the image gives 0% opacity (completely transparent) while white creates a 100% opaque surface. For this image (black in the middle, white at the edges), we need it to be the other way round. So we use the Invert option to invert the image so that black becomes opaque and white turns transparent.

- **Disable the Specular channel by clicking the checkbox to the left of Specular (so that the box becomes blank).**

We want to avoid specular highlights on the floor from the lights above. This would brighten the floor unnecessarily.

- **Drag the floor material from the Material manager and drop it onto the Floor Disc object in the Object manager.**

In this case, we don’t need to adjust the mapping or projection of the material on the object. The floor disc is a parametric object (one defined mathematically) with built-in mapping coordinates that project the texture flat from above which, as it happens, is exactly what we want.

- **Choose Render > Render View to make a quick test render.**

The floor object now has a soft edge which fades into the background.

**Summary**

In this section we learnt about reflectivity, transparency and a little about alpha channels. We used this knowledge to create a shiny floor that merged pleasingly into the background sky material. We’ll learn more about these properties later.
Bottle

Glass objects have their own special characteristics, which can be quite challenging to re-create.

The main characteristic of most glass materials is that you can see through them. To a greater or lesser degree, glass is transparent. Glass also reflects other nearby objects. In addition, light travelling from one side of the glass material to the other changes direction, causing a distortion of objects viewed through the glass – this is called refraction. Finally, you may sometimes see highlights (also known as speculars) on the outside of the glass — these are little concentrated areas of light.

So for the bottle we want to create a transparent, green material which both reflects and refracts. We’ll also adjust some highlights for extra effect.

► If you haven’t already got it open, load the scene named ‘Party Before.c4d’ from the Texturing - Basic folder in the CINEMA 4D Tutorials folder.

► In the Material manager, select New Material from the File menu. To open the Material Editor, double click on the new material that appears.

► Disable the Color channel by clicking the checkbox to the left of Color (so that the box becomes blank).

The color for this material will be defined by the transparency and reflection colors, using a color on the Color page would only brighten the bottle unnaturally. We can make the Color channel black (i.e. devoid of color) by either setting the Color RGB sliders to black or by turning off the Color channel completely.
Click the checkbox next to Transparency to enable transparency and display the Transparency page.

Use the RGB sliders to change the color to a bottle green (R=30%, G=100%, B=50%, Brightness=70%).

Enable refraction by typing a value of 1.5 into the box labelled n. (Refraction index values are denoted by the letter n.)

Setting the refractive index to 1.5, which is suitable for glass.

The 3D material preview shows the effect immediately. The glass material is now distorting anything we see behind it, just as real glass would do. The CINEMA 4D Reference manual contains a list of common refraction index values for various materials.

Enable the Fresnel option.

Watch the effect in the 3D material preview. The Fresnel option simulates an effect we see in the real world with glass, water surfaces and other transparent materials. Take a window as an example. If we stand directly in front of the window we can see right through it. If we now move to one side and look at the window from a very acute angle, we can hardly see through it at all because it mostly reflects. So the angle at which we look at a glass material defines how transparent or reflective it appears. This is called the Fresnel effect and the Fresnel option simulates this in CINEMA 4D.

Enable and display the Reflection page by clicking the checkbox to the left of Reflection.

Set the Reflection page RGB sliders to a dark green color (R=40%, G=100%, B=45%, Brightness=30%).

This gives the glass material some reflectivity. More reflection would brighten the bottle too much.

Display the Specular page by clicking on the word Specular.

This controls the highlights on the material, the extra bright reflections from lights in a scene, which are known as speculars.
Set Width to 60%, Height to 75%, Falloff to -25% and Inner Width to 25%.

The material for the bottle is green glass with strong highlights.

These settings make strong highlights with sharp edges, suitable for glass.

Drag the bottle material from the Material manager and drop it onto the Bottle object in the Object manager.

Summary

We’ve created an interesting glass material by adding transparency, reflectivity and highlights to our basic colored material. We’ll find that we will often use this combination of properties in our texturing work.
Balloons

The one thing about balloons is that they’re rubbery — now there’s a challenge.

Rubber-like materials are somewhat see-through and can be shiny or matt, depending on the actual material. They also commonly show highlights.

We will give the balloons a slightly transparent, reflective and rubbery surface. One balloon will be yellow and the other one will be pink. So once the first material is finished, we will copy it and change the color rather than recreating it all over again.

▶ If you haven’t already got it open, load the scene named ‘Party Before.c4d’ from the Texturing - Basic folder in the CINEMA 4D Tutorials folder.

▶ In the Material manager, select New Material from the File menu. To open the Material Editor, double click on the new material that appears.

▶ Change the Color RGB sliders to an orange-yellow shade (R=100%, G=75%, B=30%, Brightness=100%).

The balloon will be brightened a lot by the lighting in the scene, so we shouldn’t make the yellow too bright.

▶ Click the checkbox next to Transparency to enable transparency and display the Transparency page.

▶ Set the refraction index (n) to 1.1 (top right corner).

A balloon surface is only very slightly refractive, so we use a low refraction index.
Click on the little arrow to the right of the Image input box to open the shader popup menu.

Select bhodiNUT Channel > bhodiNUT Fresnel.

Click on the Edit... button to open the shader settings dialog.

Using the Fresnel shader, we control the amount of transparency at the edge and middle area of the balloon.

This shader is part of the bhodiNUT shader package (aka Smells Like Almonds) which is illustrated in more detail in the next chapter, Advanced Texturing with bhodiNUT shaders. The bhodiNUT Fresnel shader works just like the Fresnel option on the Transparency page, however it gives us much more control by using a gradient. We want our balloons to be completely opaque at the edges and only slightly transparent in the middle.

Under the horizontal gradient bar, double click on the left-hand color handle (also called a ‘knot’) to open the system color chooser. Change the color to black and close the system color chooser.

Remove the middle color knot from the gradient by clicking and dragging it upwards, away from the gradient.
Double click the right-hand color knot to open the system color chooser again. Change the color to light yellow (R=255 / 100%, G=232 / 90%, B=151 / 60%) and then close the color chooser.

The right-hand knot defines the central transparency.

The balloon is now transparent in the middle. The transparency color is yellow so everything behind the balloon will have a yellow tint.

Click OK to close the shader settings dialog.

Click the checkbox next to Reflection to enable reflections and display the Reflection page.

Set the Reflection RGB sliders to a very dark yellow color (R=100%, G=82%, B=62%, Brightness=30%).

We have made the material just a little bit reflective, which adds to the realism.

Display the Specular page by clicking on the word Specular.

Set Width to 100%, Height to 70% and Falloff to -22%.

This gives the surface wide highlights but with a soft falloff.

Make a copy of this material (in the Material manager, click on the balloon material then select Edit > Copy and then Edit > Paste).

The standard system copy and paste keyboard short-cuts can be used here instead.

Double click the copied balloon material to open the Material Editor.
On the Color page, adjust the RGB sliders to a pink color (R=100%, G=50%, B=60%, Brightness=100%).

Go to the Transparency page and click the Edit... button to open the shader settings dialog.

Double click the right-hand color handle to open the system color chooser again. Change the color to light pink (R=255 / 100%, G=168 / 66%, B=168 / 66%). Close the color chooser and the shader settings dialog.

Go to the Reflection page and adjust the RGB sliders to a dark pink (R=100%, G=50%, B=60%, Brightness=30%).

Now we have yellow and pink balloon materials.

Drag the yellow balloon material from the Material manager onto the Balloon object in the Object manager.

Drag the pink balloon material onto the Balloon 2 object.

Do a test render by choosing Render > Render View from the main menu.

Most of the texturing work for this scene is done now. All that’s left is to create a very simple material for the cork, a paper star for the floor and a label for the bottle.

Summary

Here we have learnt how to use the bhodiNUT Fresnel shader to give a graduated transparency (balloons are more transparent in their taut centre). We’ve also seen how to copy and paste a material, to create a variety of different balloons in our scene.
Cork

This should a pretty easy, if rather bumpy, ride.

Corks, and we’re not talking about those new-fangled plastic ones, have a bumpy surface, showing dark and light areas when illuminated. To create this effect we can use a flat, black-and-white texture as a bump map — light will react differently to the dark and light areas within this texture, giving the appearance of a coarse surface.

So, the cork needs only a very basic material using two image textures: one for the Color channel and one for the Bump channel to create a realistic relief on the surface.

- If you haven’t already got it open, load the scene named ‘Party Before.c4d’ from the Texturing - Basic folder in the CINEMA 4D Tutorials folder.
- In the Material manager, select New Material from the File menu. To open the Material Editor, double click on the new material that appears.
- On the Color page, click the Image... button and use the file dialog to open the file named ‘cork.jpg’ from the tex folder in the Texturing - Basic folder.
- Enable and display the Bump page by clicking on the checkbox to the left of the word Bump.
- On the Bump page, click the Image... button and use the file dialog to open the file named ‘cork-bump.jpg’ from the tex folder in the Texturing - Basic folder.
► **Set the Strength slider to 100%**.

The Bump channel simulates rough surfaces. In the render, brighter areas of the image used for the texture appear raised and darker areas appear depressed.

► **Disable Specular for this material by clicking the checkbox next to Specular so that the box becomes blank.**

![The cork material uses a bump map to simulate a rough surface.](image)

A cork has a rough, matt finish with no specular highlights.

► **Drag the cork material from the Material manager and drop it onto the Cork object in the Object manager.**

► **In the Attributes manager, set the Projection setting to Cylindrical.**

The Attributes manager is now displaying the projection parameters for the cork material. The cork looks more like a cylinder than a sphere, so a cylindrical projection is more suitable.

► **In the Object manager, ensure the cork material tag is still selected then select the Texture tool (in the left-hand toolbar) so that the texture reference grid is visible in the viewport.**

![Using this tool, you can see and control the texture projection for an object.](image)
In the Object manager, ensuring the cork material tag is still selected, select Fit to Object from the Texture menu.

The blue texture reference grid is much bigger than the cork and needs to be adjusted. The Fit to Object command does this automatically.

**Summary**

We’ve learnt about bump maps in this section. This is a material property that enables us to simulate depth in an object using a 2D texture. We’ve also seen how to adapt a large texture to a small object, automatically.
Paper Star

Let’s add some detail to the floor, just like laying a stencil.

We’ve already used an alpha channel to create a blurred edge to our floor, but we can also use this technique to cut out shapes in existing objects. To make the scene more interesting we want to add a paper star to the floor underneath the bottle. All we need to do for this is create a white material with an alpha channel texture in the shape of a star.

► If you haven’t already got it open, load the scene named ‘Party Before.c4d’ from the Texturing - Basic folder in the CINEMA 4D Tutorials folder.

► In the Material manager, select New Material from the File menu. To open the Material Editor, double click on the new material that appears.

► On the Color page, set the Brightness slider to 100% to make the material color white.

► Disable the Specular channel by clicking the box to the left of the word Specular so that the box becomes blank.

Paper isn’t normally shiny so we switch off the specularity.

► Click on the checkbox to the left of Alpha to enable the alpha channel and display the Alpha page.
On the Alpha page, click the Image... button and use the file dialog to open the file named ‘star.jpg’ from the tex folder in the Texturing - Basic folder.

This file is a simple black and white picture of a star. The black part of the picture will be cut out (made transparent) by the alpha channel, so this leaves us with a white star shape.

Drag the paper star material from the Material manager and drop it onto the Floor Disc object (i.e. the floor) in the Object manager.

Now the floor has two materials applied, the blue reflective material and the star on top of it. The material tag placement in the Object manager determines which material will be rendered on top, going from left (bottom-most material) to right (top-most material). Materials with alpha channels (i.e. transparent areas) should usually be on the right-hand side in the Object manager.

In the Attributes manager, set the Projection to Flat.

We can’t use the default UVW mapping because it stretches the paper star material across the whole surface of the disc. This was OK for the blue reflective material we assigned earlier, but for the paper star material we need to change the mapping to Flat so that we can manually adjust the size and position of the star.

In the Object manager, ensure the paper star material tag is selected.

Ensure the Texture tool is also selected (in the left-hand toolbar).

In the viewport we can see that the texture reference grid is perpendicular to the floor, not lying flat on it.

Select the Rotate tool (in the top toolbar) then, in the 3D viewport, click and drag the red axis, while watching the Rotation values in the Coordinates manager.
Let go of the mouse button as soon as the P value is at 90 degrees.

You can see and adjust the texture position, rotation and size from the Coordinate manager.

Alternatively, if we can’t get the exact value of 90 degrees by dragging, we can manually type it into the input box.

Also in the Coordinates manager, set all three Scale values to 165.

Increasing the Scale values makes the paper star bigger.

In the Attributes manager, on the Tag Properties page, disable the Tile option.

The Tile option in the Attributes manager switches texture tiling on or off.

We don’t want the star texture to be repeated over the entire surface of the floor, so we switch off the Tile option.

Summary

We’ve used an alpha channel to create a stencil of a star on the floor, a general principle for cutting out shapes. We’ve also seen how to position and scale a material.
Label

As a finishing touch, we’ll stick an attractive label onto our bottle.

To complete our texturing of this scene, we want to create a label for the bottle and project it cylindrically around the bottle. We will learn how to manually scale and position the label, to get it just where we want it.

- If you haven’t already got it open, load the scene named ‘Party Before.c4d’ from the Texturing - Basic folder in the CINEMA 4D Tutorials folder.
- In the Material manager, select New Material from the File menu. To open the Material Editor, double click on the new material that appears.
- On the Color page, click the Image... button and use the file dialog to open the file named ‘c4d-logo.tif’ from the tex folder in the Texturing - Basic folder.

This loads a label texture into the Color channel.

- Drag the label material from the Material manager and drop it onto the Bottle object in the Object manager.
- In the Attributes manager, set the Projection to Cylindrical.

The default projection is unsuitable for the label texture so we have to set the projection to cylindrical to be able to adjust it manually.
In the Object manager, ensure that the label material tag is still selected and then select Texture > Fit to Object. Click No when it asks if you want to include sub-objects.

This adjusts the texture to the dimensions of the bottle. A dialog asks if we want to fit the texture to the sub-objects of the bottle as well but, since it is a Lathe NURBS object with only splines as sub-objects, this is unnecessary.

Ensure the Texture tool is still selected (in the left toolbar) and then select the Scale tool (top toolbar).

In the 3D viewport, drag the green object axis down to scale down the label so that the CINEMA 4D logo does not appear distorted.

In the Attributes manager, disable the Tile option (on the Tag Properties page) to switch off tiling for this material.

In the top toolbar, select the Move tool and drag the green axis to move the label texture reference grid down slightly, as illustrated in the picture below.
From the main menu, select Render > Render to Picture Viewer to render a full sized image of the scene.

![The completed scene, time for a party!](image)

**Summary**

We should now understand that texturing isn’t just a matter of selecting an image to place on or wrap around an object surface. There are many more aspects to texturing an object which we may not have previously considered. For example, without the correct amount of reflection or bumpiness to a surface we may completely destroy the illusion that what the viewer is looking at is a photograph or movie.

Of course, we may not always want to create photorealistic imagery, this is entirely our choice. We should, however, have at least a rough idea of what we want the final images to look like before we start — there is very little point in spending time carefully tweaking how grainy a surface is if we are intending to give it all a cel-shaded, flat cartoon look.
Overview

Materials are not limited to fixed resolution 2D images. Using the built-in shaders we can create some of the most astonishing materials ever seen.

A material in CINEMA 4D combines all the information which defines a surface — color, shininess, transparency, bump and so on. These are the ‘channels’ of the material and they can have textures and/or shaders applied to them to define their surface. Materials can either be the standard CINEMA 4D materials or special materials which offer more than just the standard surface properties.

The word ‘texture’ is commonly associated with 2D images — tiff and jpegs for example. The limitation of 2D images, however, is that they are resolution dependent. So if a small texture (say 320x240 pixels) is applied to a floor object in a scene which is rendered at a resolution of 1024x768 pixels, the texture will appear blurry and not at all convincing because it will have been scaled up (or ‘resampled’ as it is known in the 2D world) from a much smaller image.

On the other hand, ‘shaders’, are resolution independent. Shaders are generated using mathematical formulas and are often referred to as being ‘procedural’. Because shaders are calculated every time we render, rather than simply scaled up or down from fixed pixel dimensions, when applied to an object shaders will always look perfect — lines and other definite structures will always look crisp no matter how close the camera gets to the surface of the object.
2D shaders act just like 2D image textures, except of course that they are resolution independent. 2D shaders are projected onto objects using standard mapping types like spherical, flat and cylindrical. 2D shaders are just as ‘flat’ as 2D image textures and need to be wrapped around the objects. 2D shaders use the U and V coordinates to decide how a flat texture is projected onto a surface. If we were to take out a huge chunk from a model textured with a 2D shader, the material would adjust itself so the same image appears in the chunk we just removed.

However, just as with a 2D image texture, if an object has a difficult shape it can be difficult to find the perfect type of projection for a 2D shader. That’s where 3D shaders come in handy. 3D shaders (which are also known as ‘volume’ or ‘volumetric’ shaders) have a real volume because they are calculated using the U, V and W coordinates, where the W is the most important part because it is this which gives the shader a volume.

Using a 3D shader means that we don’t have to worry about the projection. A 3D shader always uses UVW projection so that there is no texture stretching or smearing. 3D shaders are perfect for texturing complex shapes onto which normal 2D textures or shaders would be very difficult to project.
3D shaders have a real volume. If we cut a hole into an object we can see the shader continue inside the object.

Some shaders can be applied to the individual channels of a material, such as Color, Reflection, Bump and so on. This type of shader is called a channel shader, and they can be 2D or 3D. We might, for example, load a brick shader into the color channel of a material, a gradient shader into the transparency color channel, a translucency shader into the luminance channel, a 3D noise shader into the bump channel.

CINEMA 4D features a built-in advanced procedural shader package from bhodiNUT which adds extra functionality to the material system. The bhodiNUT shaders comprise both 2D and 3D shaders and include advanced and specialized controls to create things like brushed metal, realistic transparencies, translucency (backlighting) and much more that can’t be achieved with standard materials. The bhodiNUT shaders are available as channel shaders within the Material Editor and as standalone shader materials from the Material manager’s File menu.

The bhodiNUT shaders provide unlimited texturing possibilities and this tutorial will cover only a few of them. Our aim in completing these tutorials should be to gain enough experience to be able to experiment and create our own shader combinations. This chapter starts with an easy exercise on how to create a sky shader with clouds and then moves on to creating a more complex copper pipe material which has a layer of dirt.

**Summary**

The bhodiNUT shaders give us a wide range of procedural 2D and 3D shaders which enable us to create realistic or fantasy textures with ease. Procedural shaders, which generate textures from equations, are a powerful alternative to traditional textures created from 2D images.
Sky with Clouds

There are many ways to create a sky populated with clouds. The techniques we’ll use here to do the job have many other applications.

We are going to use a couple of the bhodiNUT SLA shaders to see how quickly and easily we can create a blue sky with some white clouds. We’ll learn how to use gradients to create a graduated sky and we’ll use a noise shader to add the clouds.

Load the scene named ‘Sky Before.c4d’ from the Texturing - SLA 1 folder in the CINEMA 4D Tutorials folder.

This scene contains a Sky object and a torus with a reflective material applied. The torus is there for the sky to reflect in, so that we can see the sky material from every angle and not only a small part.

In the Material manager, double click on the Sky material preview.

This opens the Material Editor. Our sky material will use a combination of several bhodiNUT 2D shaders for the Color channel.
Click on the small arrow button next to the image input box and select bhodiNUT Channel > bhodiNUT Gradient.

This popup menu lists all available channel shaders, including all the bhodiNUT ones. We should now be able to see a colorful gradient in the background of the viewport.

Click the Edit button to open the shader settings dialog.

The shader settings dialog is now showing a square preview of the gradient, and the actual gradient itself with small handles known as ‘knots’ below it. Each knot adds a color to the gradient.
In the popup menu at the top of the dialog, change the gradient type from 2D-U to 2D-V.

This makes the gradient go from top to bottom (V) instead of left to right (U).

Double click on the left-hand knot to open the system color chooser and set the color to white. Then close the system color chooser.

We set the sky gradient to white at the bottom to simulate the haze effect which usually makes a sky brighter near the horizon.

Set the middle knot to a light blue (R=99, G=166, B=220).

Set the right-hand knot to a dark blue (R=0, G=52, B=91).

The right-hand knot defines the color at the top of the sky.

Close the shader settings dialog by clicking the OK button and do a test render by selecting Render > Render View from the main menu.

The background is now a slight gradient from light to dark blue. But we can’t see much of the white haze.

Open the shader settings again (click the Edit button) and drag the white knot to the right until the Position input box displays a value of about 25%.

This moves the white haze up so we can see more of it in the background of our scene.

The basic gradient for the sky is done, now let’s add some clouds.
Click on the small arrow button next to the image input box and select Copy Channel to copy the gradient to the clipboard, then select bhodiNUT Channel > bhodiNUT Fusion.

The Fusion shader is very powerful. It can be used to combine an unlimited number of shaders, creating a shader hierarchy.

Click the Edit button to open the Fusion settings dialog and select Paste Channel from the popup menu next to the bottom image input box (in the Base Channel pane).

We have pasted the gradient shader so that it is now in the Base Channel of the Fusion shader; we can now mix it with another shader.
In the Blend Channel pane, click on the small arrow button next to the image input box and select bhodiNUT Channel > bhodiNUT 2D Noise.

The gradient is in the Base Channel and a 2D Noise in the Blend Channel.

We will use this 2D Noise shader to generate the clouds. There is also a 3D Noise, a volumetric shader, but as we are using spherical mapping to project the sky material onto the inside of our big sky sphere, we don’t need the volumetric noise.
In the Blend Channel pane, click the Edit button to open the 2D Noise settings dialog.

In this dialog we can change parameters like noise color (it doesn’t have to be a black and white noise), contrast, scale (noise can be stretched horizontally or vertically), the noise type and more.

In the Noise pane, set the noise type to Wavy Turbulence.

This is a different type of noise, more interesting than the default noise, perfect for clouds.
In the Scale pane, Set the U Scale value to 500% and the V Scale value to 200%.

![Scaling up to get larger clouds.](image)

We have made the noise structure bigger, which will result in bigger clouds, and we have also stretched it sideways.

In the Clipping pane, move the Low Clip slider to 53% and the High Clip slider to 73%.

![Use the Low Clip and High Clip sliders to generate white clouds.](image)

This narrows the value range of the noise, making dark areas even darker and bright areas even brighter. So instead of a foggy and overcast sky we now have a few white fluffy clouds.

Close the 2D Noise settings dialog by clicking the OK button to return to the Fusion dialog.

On the left in the Fusion dialog, set the Mode to Screen.

The clouds are now mixed with the gradient. The Screen mode blends them into the gradient, depending on the brightness of the clouds.

Close the Fusion dialog by clicking the OK button and choose Render > Render View from the main menu.

Summary

We could additionally control the visibility of the clouds with the Blend slider in the Fusion shader, or we could try other blend methods to get different results. The blend modes work just like mixing layers in a 2D graphics application. Instead of a gradient and a noise shader, we could combine the gradient with another Fusion shader, which again includes two shaders which are mixed. Fusion lets us mix anything with anything. It is not limited to shaders only, we can even mix a shader with an image. It is one of the most powerful shaders in the bhodiNUT shader package.
Dirty Copper

*bhodiNUT’s SLA shaders can be used to simulate a wide range of materials. Here we’ll use one of the 3D shaders to age a piece of copper.*

The bhodiNUT volume shaders all have custom interfaces. Some elements, like Specularity, Roughness and Reflection controls, are present in all of the 3D shaders, but some others are specially designed for a specific 3D shader.

BANJI can be used for glass and translucent surfaces, BANZI for wood, CHEEN for electron microscope type materials, DANEL for metallic surfaces, including effects like blurry reflection and streaking anisotropic highlights, MABEL for marble surfaces and NUKEI for surfaces which actually consist of two surfaces, like blistering paint or rusty metal.

NUKEI can also be used to create surfaces which have a layer of dust or dirt on top of them, which is what we will be doing on the next few pages. We will create a copper pipe material with a dirt layer on top. This dirt layer will be rendered only on those parts of the object which face upwards. Our shader will be a combination of NUKEI, Falloff and the Fusion and Noise shaders.

► Load the file named ‘Dirty Copper Before.c4d’ from the Texturing - SLA 2 folder in the CINEMA 4D Tutorials folder.

This scene contains a sphere and a pipe shaped object.
In the Material manager, select File > bhodiNUT Volume > bhodiNUT NUKEI.

The Material manager menu is where the volumetric 3D shaders are listed.

Assign this new material to the Sphere and the Sweep NURBS objects by dragging it from the Material manager and dropping it onto each object in turn in the Object manager.

In the Material manager, double click the material preview to open the NUKEI dialog. Go to the Diffuse page by clicking on the word Diffuse in the channels list on the left-hand side of the dialog.

This section defines the overall color of the first surface (the copper).
Use the Diffuse Color RGB sliders to set the color to orange (R=100%, G=60%, B=40%).

Go to the Specular 3 page and set the RGB sliders to the same values as the diffuse color.

We have set the diffuse color to orange, and we need to set the specular highlight color to orange as well or we would get a copper material with (by default) blue highlights. The bhodiNUT 3D shaders have three specular highlight controls instead of one. This provides a lot of extra control compared to a normal CINEMA 4D material, which has just the one specular channel. Each specular channel has a default setup and the first two are already right for our purposes here, so we’ve just tweaked the third channel.

Go to the Fusing page and load the bhodiNUT 3D Noise shader into the image input box. Click the Edit button to open the 3D Noise settings dialog.

Set the Global Scale to 500% and close the dialog.

The second surface (the dirt) is now visible in the small preview.

In the NUKEI dialog, locate the popup just above the channels list and set it to Surface 2.

Changing from Surface 1 to Surface 2 means we are now editing the parameters of the second surface, the dirt.

Go to the Diffuse page and set the Brightness slider to 0%.

The dirt surface is now completely black. That’s all we need to change for the dirt.

Go to the Fusing page and load the bhodiNUT Fusion shader into the image input box (replacing the 3D Noise). Click the Edit button to open the Fusion dialog.

The 3D Noise alone won’t make a dirt layer which appears only on the upwards facing parts of our objects. We need to use the Fusion shader to combine the 3D Noise with a special shader which controls where the 3D Noise will be visible.

In the Fusion dialog, load the bhodiNUT 3D Noise into the image input box for the Base Channel.

Into the Blend Channel, load bhodiNUT Falloff and click the Edit button to open the Falloff dialog.

This Falloff shader is similar to the Gradient shader we used in the sky tutorial. With the Falloff shader, the gradient is projected onto the object surface depending on which direction the object surface is facing. In our case, we want dirt to appear on areas facing upwards and no dirt on areas facing downwards. Anything between will be a falloff from dirt to no dirt.
Drag the middle knot of the gradient upwards to remove it.

Move the left-hand knot to a position of about 60%.

Double click the right-hand knot and use the system color chooser to set its color to white.

The Falloff shader can make dirt appear only on the upwards facing areas of an object.

Close the Falloff dialog.

In the Fusion dialog, set the Mode to Levr and Blend to 75%.

Levr is a special blending mode which adds the noise structure to the gradient using a hard and jagged transition; the Blend value controls how hard this transition is.

Enable the Invert Output option.

As we have set it up, the dirt will be visible in the white part of the material preview image, so at the moment it is the wrong way around. Enabling the Invert Output option reverses this so that we don’t need to change the gradient itself.

In the Base Channel pane, click the Edit button to open the 3D Noise dialog. Set the Global Scale to 30% and close the dialog.

This adjusts the scale of the dirt graininess. A smaller scale will look more convincing.

Close the Fusion dialog.

We can now see the black dirt on top of the sphere in the little NUKEI preview.
Close the NUKEI dialog and select Render > Render View from the main menu.

On top of the objects is now a thick, black layer of dirt.

We now have a thick, black layer of dirt on top of our shiny copper pipe. To control the thickness of the dirt, all we need to do is change the gradient in the Falloff shader. For example, moving the black knot more to the right will result in less dirt.

**Summary**

The bhodiNUT shaders are phenomenally powerful and experimentation is one of the best ways to learn your way around them. Try playing with the Fusion shader to start with. Fusion takes two materials and combines them — to create a tiled floor, for example, you might combine a noise shader for the bumpiness of the tiles with the tile shader to create the beveled edges.
CINEMA 4D

Lighting
3-Point Lighting

The 3-point lighting method is a common way to illuminate either a single object or several objects in a scene, which is what we’ll do in this tutorial.

As its name suggests, 3-point lighting uses three lights: a key light, a fill light and a backlight. The key light is the most dominant light in the scene and defines the main lighting angle. The fill light is used to brighten the areas which are unaffected by the key light and which would otherwise remain completely black. Giving the key light a yellow or orange color makes the lighting in the scene appear warm. The backlight adds a bright rim around the objects to make the contours visible and to make the objects stand out from the background. Blue is a suitable color for the backlight, as cooler colors are seen by the human eye as somewhat more distant than warmer colors, thus giving more apparent depth to your scene.

The picture below illustrates the common positions of key, fill and backlight. The key light is to the left of the camera, the fill light to the right and the backlight is cast from a bigger distance behind.

Note that the lights used for a 3-point lighting setup are meant for illuminating the main object in our scene. Objects in the background need their own, separate lighting setup.
The following pictures illustrate the difference between the CINEMA 4D default lighting (one white light source directly next to the camera) and the 3-point lighting method. While with the default lighting the object looks flat and uninteresting and much of the geometry detail gets lost, with 3-point lighting it has more depth and detail. It generally looks more appealing.

However, 3-point lighting is not a strict rule, it is more of a good starting point when setting up the lights in a scene. Since every scene and every object is different, the 3-point lighting needs to be adjusted every time you apply it to a scene. Different lighting angles produce different lighting moods. Extreme angles can be useful to create a dramatic mood in a scene. The picture below shows the effects of different key light positions.
1 = key light positioned left of the camera and slightly higher. 2 = key light positioned above the head. 3 = key light positioned below the head. 4 = key light positioned behind the head.

Load the scene named ‘Still Life Before.c4d’ from the Lighting - 3 Point folder in the CINEMA 4D Tutorials folder.

This scene contains a still life. The objects, materials and camera are already set up in this still life scene.

Select Render > Render View to make a preview rendering of the scene.

This is what the still life looks like with the default CINEMA 4D lighting.

When there are no light sources in a scene, CINEMA 4D automatically uses the default lighting for rendering. This lighting uses an internal omni light which is positioned slightly left of the camera. This light is completely white and doesn’t cast any shadow. As we can see, the objects appear flat and their contours are invisible because they blend into the black background. Even the floor, which is actually blue, appears black because the light hits the floor at an angle which is too oblique to illuminate it.
Select Objects > Scene > Light from the main menu to create the first light and name it Key.

This will be our key light. As soon as we create the light we can see the lighting change in the viewport. Note that lighting in the viewport is shown only if the Gouraud Shading mode in the View menu of the view is active. Quick Shading always uses the standard lighting, which is useful if we have many lights in our scene and want to speed up the display.

In the Object manager, ensure that Key is selected then hold down the right mouse button and in the viewport move the mouse to the right until all objects are visible.

This moves the light towards us. Dragging with the left mouse button moves left, right, up and down while dragging with the right mouse button moves an object away from or towards us.

Hold down the left mouse button and in the viewport drag upwards and left.

The objects should now be illuminated from the top left.

In the Object manager, ensure the Key light is selected and click the Coord. tab in the Attributes manager.

In the Attributes manager, set the position of this light to P.X=1.74, P.Y=376 and P.Z=-795.

These are the position values for the key light.

The key light illuminates the scene from the side.
This positions the key light just above the objects, illuminating the scene from the side. Since we all want to end up with the same result, in this step and the next few steps we are positioning the lights using exact numbers. Normally we would move each light around, do some test renders and adjust the lights until the lighting is perfect. For this scene, we will use the light position values that we think make a perfect lighting for this still life.

▶ Create a second light and name it Fill.

▶ In the Object manager, ensure that the Fill light is selected then click the General tab in the Attributes manager.

▶ Use the RGB sliders to make the light orange (R=100%, G=83%, B=68%, Brightness=100%).

Making the fill light orange adds a bit of color to the scene and generally makes it appear warmer, a bit like the light from a flame would.

▶ In the Attributes manager, click the Coord. tab and set the Fill light position to P.X=1178, P.Y=159 and P.Z=187.

These are the position values for the fill light.

The fill light illuminates the scene from the opposite side to the key light.

This places the light to the right of the objects, brightening those areas that are left dark by the key light. The fill light is also slightly lower than the key light, illuminating more from below.

▶ Create a third light and name it Back.

This is our backlight.
In the Object manager, ensure that the Back light is selected then click the General tab in the Attributes manager.

In the Attributes manager, use the RGB sliders to change the color of this light to a cold blue (R=53%, G=77%, B=100%, Brightness=200%).

Blue is a cool color and, when used as a light from behind, it gives the impression of depth to a scene. It also visually implies that there is something else around the scene since cold blue light is often associated with light cast by the sky. Note that the brightness of a light is not limited to 100%. We have set the backlight to 200% brightness because it will be visible only at the outer edges of the objects and would hardly be visible at all with 100% brightness.

In the Attributes manager, click the Coord. tab and Set the Back light to a position of P.X=-507, P.Y=2135, P.Z=2005.

These are the position values for the backlight.

The backlight illuminates the scene from above and behind.

This moves the backlight far away, illuminating the top areas of the objects from behind. The backlight also provides the main lighting for the floor and makes the objects stand out from the background.

In the Object manager, Shift-click on all three lights to multi-select them.
In the Attributes manager, click the Details tab and set the Contrast to -30%.

The Contrast value controls the softness of the edges between illuminated and non-illuminated areas.

Left: Contrast = 0%. Right: Contrast = -30%.

This reduces the contrast for all three lights. The contrast value controls the softness of the transition edge between areas that are illuminated by a light and those that are not. Under certain circumstances, when a surface is lit by more than one light, unnatural seams can become visible. This is the case for the egg in our scene because the falloff between the brighter and darker areas is too abrupt. A negative contrast value makes the falloff softer and disguises the seam effect.

In the Object manager, multi-select the Back and Key lights and click the General tab in the Attributes manager.

In the Attributes manager, set the Shadow to Soft.

Select Render > Render View from the main menu to do a quick test render.

The backlight is now casting the main shadow from behind and the key light casts a shadow from the side. However, the shadows on the floor and on the carrot are much too dark, they don’t look at all realistic. We never find absolutely dark shadows in reality, there will always be indirect light bouncing off walls or other objects brightening the shadows.

In the Object manager, multi-select the Back and Key lights and click the Shadow tab in the Attributes manager.
In the Attributes manager, set the Density to 50%.

This makes the shadows cast by both lights half as dark.

In the Object manager, multi-select all three lights and click the General tab in the Attributes manager.

In the Attributes manager, change the Type to Spot.

Omni lights cast their light in all directions. They don’t only illuminate the main objects in the scene, they also illuminate the surroundings, in our case the floor. We have set all our lights to be spot lights so that they illuminate only a small area around the still life. The objects will stand out more if the surroundings are dark. It gives our scene a better contrast.

In the Attributes manager, click the Details tab and set the Inner Angle to 20°.

A spot light has two cone angle settings, an inner cone angle and an outer cone angle. Within the inner cone, the light intensity is always 100% and between inner cone and outer cone the light falls off from 100% to 0% intensity. As all three lights are still multi-selected in the Object manager, here we are changing the Inner Angle of the cone for all three lights.

In the 3D viewport, click the Toggle View button (top right corner) to switch to the 4-view mode.

The spot lights are all pointing in the wrong direction at the moment so we will use one of the viewports as a lighting view from where we can adjust the direction and cone angles for each light.
In the Object manager, select the Key light and from the Cameras menu of the bottom-left viewport select Link Active Object.

A light can act as a camera, enabling us to see at what it is pointing.

The key light is now acting as a camera and we can see where it is pointing. We can also see two circles, the inner and the outer cone.

Select the Rotate tool (top toolbar) and rotate the Key light until all objects are visible in the middle of the viewport (Rotation H = -8.6°, P = -15.2°, B = 0°).

We may have to use other viewports to rotate the light to prevent it from tilting sideways or turning upside down.

Drag one of the orange handles for the inner cone until it surrounds all objects. Do the same for the outer cone, so that there is a small distance between inner and outer cone. (Inner Angle = 40°, Outer Angle = 50°)

The outer cone needs to be slightly bigger than the inner cone to give the spot light a soft falloff.

To give the spot light a soft falloff, we need to make the outer cone a bit bigger than the inner cone.
In the Object manager, select the Fill light and from the Cameras menu of the bottom-left viewport select Link Active Object.

Select the Rotate tool and rotate the Fill light so that it points towards the still life. (Rotation H = 84°, P = -4.7°, B = 0°)

Adjust the cone angles of the Fill light to make the inner and outer cones surround all objects. (Inner Angle = 49°, Outer Angle = 60°)

In the Object manager, select the Back light and from the Cameras menu of the bottom-left viewport select Link Active Object.

Select the Rotate tool and rotate the Back light until it points at the objects. (Rotation H = -158.9°, P = -48.6°, B = 0°)

Adjust the cone angles of the Back light to make the inner and outer cones surround all objects. (Inner Angle = 23.7°, Outer Angle = 42.9°)
Select Render > Render to Picture Viewer to render the scene.

This is the final still life render.

Now the still life has a lighting scheme which makes the whole scene more colorful and interesting to look at.

Summary

This 3-point lighting setup can be applied to more complex scenes just as well as this one. If we have other objects in the background but want to emphasize a few objects in the foreground, then a good setup is a warm and colorful 3-point lighting for those main objects in the foreground and a darker, cooler blue lighting for the background and the surrounding objects.

Always remember that lighting is just as much a part of a scene as the objects. Striking shadows and dramatic colors can turn dull scenes into masterpieces. Some of the best pieces of art use a very simple scene and then generate interesting areas through the careful use of light and shadows.

However we choose to illuminate our projects, we should avoid using the default single light situated just next to the camera, as it will always give dull, uninteresting results.
Volumetric Lighting

*CINEMA 4D* provides us with a vast range of lighting options. Using volumetric lights we will create an impressive corporate logo.

On TV we often see the effect where a logo or title casts visible rays of light. This effect looks impressive and is actually very easy to produce. In this tutorial we are going to learn how to use volumetric lighting to achieve such an effect.

In reality, visible light is lots of small particles of dust or water which are illuminated by the light in a dusty or foggy environment. In *CINEMA 4D* we can make lights visible, even without the fog, by simply enabling the visible light option. If we want to block visible light, or parts of it, then we need to use volumetric lighting. Volumetric lighting can also be reversed — instead of blocking parts of a visible light cone, objects can be placed inside the light cone to make them appear as if they are casting light from their own surface. This is called inverse volumetric lighting.
Load the scene named ‘Volumetric Light Before.c4d’ from the Lighting - Volumetrics folder in the CINEMA 4D Tutorials folder.

This scene contains a simple logo, a background and lighting set up. All we need to add is the effect.

From the main menu, select Objects > Scene > Target Light.

The target light is nothing more than a normal spot light with a target null object and a target expression. The target expression makes the light always point towards the target object. The target object and the logo are both placed at the world origin so the spot light is already pointing in the correct direction.

Use the Move tool or the Coordinates manager to adjust the position of the spot light so that it is behind the logo. (X=0, Y= -170, Z=1112)

The spot light needs to be positioned behind the logo (here seen from the side).

In the Attributes manager, click the General tab and adjust the color of the spot light to a very bright blue. (R = 80, G = 90, B = 100, Brightness = 300)

The color and brightness settings are not only used for the light that is cast onto objects, they also influence the brightness of the visible light. We want the light rays to be very visible so we have set the brightness to 300%.

In the Attributes manager, set Visible Light to Inverse Volumetric.

This is the option which is mainly responsible for the effect we want to achieve.
Click the Visibility tab in the Attributes manager.

On this page we can further adjust the visible light.

Set the Outer Distance to 4000 and the Inner Distance to 3000.

Our scene is quite large and the default length (Outer Distance) of the light cone is too short, it doesn’t even reach the logo object. So we must set the Outer Distance to a much larger value and also use the Inner Distance value to make the light cone 100% intense over a distance of 3000 units; after 3000 units the visibility will fall off until it reaches 0% intensity at a distance of 4000 units.

Activate the 3D viewport by clicking on its title bar and select Render > Render View from the main menu.

This effect is especially impressive when animated. On the CD, in the same folder as this scene, you will find a rendered movie which demonstrates what it looks like if the light is moved up and down behind the logo.

Summary

Volumetric shafts of light are great for creating an atmosphere and giving a scene some sense of feeling. A single shaft of light can be seen as an escape, or a glimmer of hope, whereas many shafts of light can be used to portray something amazing or powerful. Here, we have used volumetric light to suggest energy escaping from the logo, giving the impression of a powerful force. Other places you might find volumetric light useful include dusty attics, smoky bars and light streaming through church windows — when used in such a way, the volumetric light will take on the colored properties of the stained glass.
Keyframes 1

Animating Position

Our first look at simple animation techniques will explore the use of keyframes to animate the position of an object.

There are many ways of creating an animation within CINEMA 4D but the simplest is to record the different states of an object at different times and let CINEMA 4D produce an animation by interpolating between each state.

Let’s explain that more simply. Say we want to animate a car moving along a road which is mostly straight but has a couple of bends. We can record the position and direction of the car at the beginning of its journey; this information will be recorded in a keyframe.

Since the car is moving along a straight road to start with, we don’t need to record another keyframe until the start of the bend, when the direction of the car starts to change. CINEMA 4D can work out the car’s position and direction between these two keyframes by interpolation — intelligently guessing the car’s state between the two keyframes. We then will want to add more keyframes as the car goes round a bend to help CINEMA 4D work out its position.

So, first of all, we will see how to record keyframes of an object’s position to produce a simple animation.
Click the top icon in the left-hand toolbar and select Animation.l4d.

There are several layouts supplied with CINEMA 4D, one of them is designed for use while animating. Windows like the Timeline and F-Curves managers are open while those which are not necessary for animation have been closed or relocated.

To keep things simple, all we want to do is make an object move from A to B to C.

Create a new scene by selecting File > New from the main menu.

Select Objects > Primitive > Sphere from the main menu, or by choosing it from the Primitives command group in the top toolbar.

Select the Move tool and in the viewport drag the sphere to the top left corner of the scene.

This is the starting position.

In the animation toolbar (below the viewports area), disable the buttons for Scale, Rotation and Parameter.

In the animation toolbar, we deactivate everything except the Position button. At this point, we only want to make the object move.
Click the Record button.

The first red dot is the Record button.

This records the sphere's position in a keyframe at the beginning of our short animation.

Still in the animation toolbar, drag the time slider to frame 45.

Ensure the Move tool is still selected and in the viewport drag the sphere towards the middle and bottom of the scene.

Click the Record button again.

This records the sphere at its new position in a keyframe at frame 45. We can now drag the time slider back and forth (known as 'scrubbing') to see the sphere move in the viewport.

Drag the time slider all the way to the right.

Ensure the Move tool is still selected and in the viewport drag the sphere to the top right corner of the scene.

Click the Record button again.

This records the new position of the sphere in a keyframe at the end of our animation.

Press the Play button to see the full animation.

The sphere now moves from A to B to C.
Select File > Save and select a suitable location and filename.

We will continue working with this scene in the next parts of this Keyframe Animation chapter.

Summary

We have seen that we can record keyframes of an object’s position by simply hitting the Record button within the CINEMA 4D interface. We can then move the object and hit Record again. CINEMA 4D will create an animation by interpolating between these keyframes. We can animate cameras and lights in exactly the same way; already we can see the power of such a system.
Keyframes 2
Animating Size

Another object property that we can easily animate is size. Here we will scale a sphere to produce a pulsating blob.

We can animate the size of an object by recording keyframes at different sizes and CINEMA 4D will interpolate between the keyframes. Animating size allows us to create anything from a deflating balloon to a beating heart.

► **If you don’t already have it open, load the scene we saved at the end of the previous tutorial.**

In this scene we set up a short animation that moved a sphere from A to B to C. Animating the size of an object is achieved in almost exactly the same way as animating its position.

► **Ensure that the Object tool is selected in the left-hand toolbar.**

While animating the scale of an object, ensure that the Object tool is selected.

There is an importance difference between the Object tool and the Model tool. To learn more about this difference, please see the Animation section of the Tips and Tricks chapter at the end of this manual.

► **Drag the time slider back to the frame 0.**

► **In the animation toolbar, disable the Position button and enable the Scale button.**

To record only size changes, enable the Scale button.
Select the Scale tool in the top toolbar.

To scale objects, select the Scale tool in the top toolbar.

In the viewport, drag the green object axis to the left to flatten the sphere.

Flatten the sphere by dragging the green object axis.

In the animation toolbar, click the Record button.

We have now stored this shape in the keyframe at frame 0.

Drag the time slider to frame 45.

Ensure that the Scale tool is still selected and in the viewport drag the green object axis to the right to stretch the sphere.

In the animation toolbar, click the Record button.

We have stored this new shape in the keyframe at frame 45. If we have done everything correctly, the sphere now squashes and stretches over the first half of the animation. We can check this by scrubbing the time slider back and forth.

Drag the time slider all the way to the right.

Ensure that the Scale tool is still selected and in the viewport drag the green object axis to the left to squash the sphere again.

In the animation toolbar, click the Record button.

We have stored the new shape in the animation’s last keyframe.

Press Play to view the animation.

While the sphere moves across the screen, it now squashes and stretches and squashes again.

Select File > Save and select a suitable location and filename.

We will continue working with this scene in the next parts of this Keyframe Animation chapter.
Summary
As we have seen, animating the size of an object uses the same principles as animating its position; we use keyframes to record particular sizes and CINEMA 4D interpolates between these sizes. Animating size is not often used but can be employed for some interesting effects, such as a character’s blinking eyes.
Keyframes 3
Animating Rotation

As well as introducing the animation of a rotating object we’ll look at automatic keyframing, a useful and time-saving tool.

As we would expect, in addition to position and size we can also animate the rotation of an object. This is something that we will often want to do, to produce anything from a spinning logo to a ship’s propellor. For this tutorial we will also see how we can make the process of recording keyframes a bit easier and quicker. We will use ‘automatic keyframing’ which makes clicking the Record button unnecessary. As its name suggests, automatic keyframing records any changes automatically.

► If you don’t already have it open, load the scene we saved at the end of the previous tutorial.

In this scene we have set up a short animation that moves a sphere from A to B to C while at the same time squashing and stretching and squashing again. Now we are going to add some rotation to the sphere.

► Ensure that the Object tool is selected in the left-hand toolbar.

When animating, always ensure that the Object tool is selected.

► Drag the time slider back to frame 0.

► In the animation toolbar, disable the Scale button and enable the Rotate button.

To record rotation, enable the Rotate button.
Still in the animation toolbar, enable the button with the red dot and white key (to the right of the Record button).

This button enables automatic keyframing. This button switches on automatic keyframing. To make it obvious to us that automatic keyframing is enabled, a red border is drawn around the viewport.

When using automatic keyframing, we should remember to switch it off when we are finished animating to avoid recording unwanted changes.

Select the Rotate tool in the top toolbar and in the viewport drag outside the yellow rotation circle to rotate the sphere down a bit on its right side.

This new rotation angle is automatically recorded in the keyframe at frame 0.

Drag the time slider to frame 45.

In the viewport, rotate the sphere so that it is horizontal again.

This new rotation angle is automatically recorded in the keyframe at frame 45.

Drag the time slider all the way to the right.

In the viewport, rotate the sphere down a bit on its left side.

This new rotation angle is automatically recorded in the final keyframe of the animation.

In the animation toolbar, press Play to view the animation.

Now the sphere not only move across the screen and changes shape, it also rotates sideways.
Summary

Now that we have seen how to record keyframes manually and automatically, we should be set up for most animation projects. Where possible, we should make use of the automatic keyframing option as this will record all changes we make to a scene — if we decide to record keyframes manually we risk either missing an important change or having to record absolutely every parameter, which can take a while for a complex movement.
The Timeline and F-Curves

When it comes to realistic animation, as with so much in life, the devil is in the detail. However, using CINEMA 4D’s powerful Timeline and intuitive F-Curve manager, we don’t have to go to Hell and back first before an animation does exactly what we want it to!

It is said that 90% of any project, whether animation, computer programming or building, is spent fine tuning everything to finish it off. A relatively short amount of time is spent putting together the meat of the work. Once this is done, we then spend most of our time adjusting everything so that it flows and generally looks as we want it to. Anything we can do to shorten this long part of the production process will help greatly.

F-Curves, or Function Curves to give them their full name, are used to fine tune exactly how one object gets from one position to another. F-Curves aren’t just limited to adjusting the position, scale and rotation of an object though, they also enable us to adjust how an attribute of an object changes over time.

F-Curves are displayed visually on a 2D grid. Time is shown along the bottom horizontally, while the actual value is displayed on the left. By adjusting the curve we can alter the speed of change and its intensity.

In this tutorial we will be animating the propellers of an aeroplane. We will animate one propeller using the Timeline and adjust its rotation in the F-Curves manager. Then we will copy the animation to the second propeller.

Load the scene named ‘Propeller Plane Before.c4d’ from the Animation - FCurves folder in the CINEMA 4D Tutorials folder.

This scene contains a model of an aeroplane. We want to animate the propellers.
Ensure the time slider is at frame 0.

In the Object manager, select Propeller1.

Select Propeller1.

In the Attributes manager, click the Coord. tab to display the Coordinates page.

Command-click on a Mac if you have only one mouse button.

Right-click on the letter P in R.P and select Animation > Add Keyframe from the context menu.

Record a keyframe for the P rotation value.

R stands for Rotation and P stands for Pitch. This is the rotation axis around which we want the propeller to rotate. At the moment it is set to 0°.

As soon as we have added the keyframe, a red dot is displayed next to R.P. This red dot means that there is a keyframe for this parameter at the current frame. (A red circle would mean that there is an animation track but no keyframe on this frame of the animation.)

Note that there is a difference between right-clicking on the P and the R in R.P. Clicking on the R means that all three rotation parameters get selected, while clicking on P selects only the Pitch rotation.

Command-click on a Mac if you have only one mouse button.

Right-click on the letter P in R.P again and select Animation > Show Track from the context menu.

The Timeline shows the animation track.

The animation track is now displayed in the Timeline. The small rectangle at the beginning of the track is the keyframe we added. Alternatively, we could have dragged the Propeller1 object from the Object manager into the Timeline, which would have had the same result.
In the Timeline, select the keyframe at the beginning of the animation track by clicking once on it.

A bright red outline around the keyframe tells us that it is selected. Note that if we want to select more than one keyframe, we can either hold down Shift to multi-select them or click into an empty area and drag a box around the keyframes. If we hold down Shift we can use the box selection method multiple times until we have selected all the keyframes we need.

Hold down the Ctrl key and drag the keyframe to the end of the track until ‘Selection: 360 F’ is displayed at the bottom of the Timeline window, then release the mouse button.

This places a copy of the keyframe at the end of the animation. (Ctrl-dragging makes a copy of something.) The copied keyframe is now also selected and the keyframe information and values are displayed in the Attributes manager.

If you forget to hold down the Ctrl key, which would move the key instead of copying it, you can always cancel the process by pressing the Esc key before you let go of the mouse button. The key will then jump back to its original position without being moved.

In the Attributes manager, type 360*20 into the Value input box.

Every numerical input box can perform math operations.

We want the propeller to rotate 20 times over the length of the animation, which would be 20*360 degrees (the star between the numbers means ‘times’ or ‘multiplied by’). Don’t worry, there is no need to get a calculator, CINEMA 4D can do the calculation for us. Every numerical input box can perform mathematical operations so we simply tell CINEMA 4D to calculate 20*360 for us (which is 7200).

In the animation toolbar, Press the Play button.

The propeller is now rotating at a constant speed. There is no acceleration or deceleration at all.
Command-click on a Mac if you have only one mouse button.

In the Object manager, right-click on the Propeller1 object and select Show F-Curves from the context menu.

In the F-Curve manager, a straight line means a constant rate of change.

This command opens the F-Curve manager. Alternatively we could have dragged the object from the Object manager and dropped it into the F-Curve manager. We don’t want the propeller to rotate at a constant speed, we want it to accelerate and decelerate. This is why we need to adjust the F-Curve for the propeller rotation. The F-Curve for our rotation track is a straight line at the moment, and this tells us that the speed at which the parameter (the rotation) changes is constant.

In the F-Curve manager, select Edit > Select All.

Both keyframes on the curve are now selected and displayed in red.

Select Curves > Custom Tangents > Soft Interpolation.

This command creates tangents for the curve points which we can manipulate. These tangents work just like spline point tangents. Short tangents result in hard corners while long tangents make a very round curve.
Select Curves > Custom Tangents > Flat.

The tangents are now flat and our curve looks different. It eases in at the beginning, described visually by the shallow incline that gradually gets steeper, and eases out at the end as the curve gets less steep once again.

In the animation toolbar, press the Play button.

The speed of the propeller rotation has changed noticeably. It starts slow, accelerates and then decelerates before it stops. However, we think the stop is a bit too abrupt.

Use the zoom icon in the top right-hand corner of the F-Curve manager to zoom out and make some space in the chart to the right of the F-Curve.
Drag the handle for the right-hand curve point tangent to the right until it is roughly twice as long as before.

A longer tangent results in a bigger rounding.

The longer a tangent, the bigger the rounding in the curve. By making the tangent longer we make the easing out of the curve even more noticeable.

In the animation toolbar, press the Play button.

The propeller rotation is now taking more time to decelerate than it takes to accelerate and the rotation looks realistic.

Open the Timeline by clicking on its tab.

In the Object manager, drag the Propeller2 object and drop it into the Timeline.

The Propeller2 object is now listed in the Timeline as well.

In the Timeline, Ctrl-drag Propeller1 onto Propeller2 and let go of the mouse button when the pointer displays a little + symbol.

Copying the animation from one object to another.

We have now copied the animation from the first propeller to the second. Note that Ctrl-dragging copies things. If you want to move something, drag without holding down Ctrl. Dragging or Ctrl-dragging an object name in the Timeline moves or copies all animation tracks for that object. It is also possible to move or copy individual animation tracks by dragging or Ctrl-dragging tracks instead of object names.
► **In the animation toolbar, press the Play button.**

The second propeller rotates just like the first one. We think this might be just a little too synchronized.

► **In the Timeline, drag the first keyframe for Propeller2 to the right until ‘Selection 20 F’ is displayed at the bottom of the Timeline window.**

This offsets the start of the second rotation by 20 frames.

► **In the animation toolbar, press the Play button.**

Propeller2 is now starting slightly later than Propeller1. But still, the propellers start and end with exactly the same rotation values which looks strange to us, possibly because in reality it would hardly ever happen.

► **Open the F-Curve manager.**

► **In the Object manager, drag the Propeller2 object and drop it into the F-Curve manager.**

The F-Curve for the rotation of Propeller2 is now displayed.

► **In the F-Curve manager, select Edit > Select All.**

Both curve points (keyframes) are selected.

► **Hold down the mouse button on one of the two curve points and drag the whole curve upwards by a value of roughly 35.**

You may have to zoom in on the first curve point to be able to offset the curve over such a small distance.

► **In the animation toolbar, press the Play button.**

The second propeller is now starting and ending with a different rotation value than the first one. The animation doesn’t look synchronized any more, which makes it much more believable.
Summary
We have learned how to use the Timeline and how important F-Curves are for realistic motion. However, F-Curves don’t only control rotation or movement, they can control any parameter. For example, if we animate the brightness of a light we can use the F-Curve to make it brighter very slowly and then darken again quite abruptly. By adding more keyframes between and changing the curve to look like a waveform, we can even make the light flicker. In short, we will find that F-Curves reduce the number of keyframes we need, thus making adjustments easier and cutting production times.
Animating Parameters

Animation isn’t just restricted to moving objects. You can animate virtually any parameter of any object, leading to literally endless possibilities and combinations.

Animation in CINEMA 4D is not limited to position, scale and rotation. Almost any parameter can be animated. This tutorial will be illustrating one example of parameter animation: melting an object and then changing the material from red clay to shiny metal. As we proceed through this tutorial, we should note how many different parameters there are for each object. If a parameter has no small x to the left of its name in the Attributes manager, then we may animate it.

Load the scene named ‘Parameter Anim Before.c4d’ from the Animation - Parameters folder in the CINEMA 4D Tutorials folder.

This scene contains a red clay sphere.

In this scene you will find a sphere covered with a red clay material.
Select Objects > Deformation > Melt from the main menu to create a melt deformer.

In the Object manager, drag the Melt deformer and drop it into the Sphere object.

Select Objects > Deformation > Melt from the main menu to create a melt deformer.

Select the Move tool and in the viewport drag the Melt deformer to the bottom of the sphere. (Y = -100)

The Melt deformer needs to be inside the Sphere object.

The Melt deformer needs to be at the bottom of the melting object, otherwise the geometry gets cut off at the bottom when melting. If we now drag the orange handle back and forth we can test the melt effect.

Ensure the time slider is at frame 0 and then, in the Object manager, select the Melt deformer.

In the Attributes manager, on the Object tab, set the Strength to 0%.

In the viewport, the strength is represented by the orange handle.

Right-click on the word Strength and select Animation > Add Keyframe from the context menu.

This records the starting position for the melt effect.

Drag the time slider to the last frame of the animation.

In the Attributes manager, set the Strength to 100%.
Command-click on a Mac if you have only one mouse button.

Right-click on Strength again and select Animation > Add Keyframe from the context menu.

This records the end position for the Melt deformer.

This records the melt effect in its end stage, meaning the object is melted completely flat on the ground.

Drag the time slider back to frame 0.

In the Material manager, click once on the red clay material preview to display the material parameters in the Attributes manager.

The Attributes manager lists all material parameters.

The sphere has a red clay material assigned. We want to morph this into a shiny metal while the sphere is melting by animating several of the material parameters like color, specularity and reflectivity. The material is now selected and the Attributes manager displays all parameters of the red clay material. Although we could also use the Material Editor to make changes to the material, the Attributes manager is the only place where we can animate all these parameters.

Note that if there is an x symbol displayed in front of a parameter name it cannot be animated.
► Enable reflections for this material by first clicking the Basic tab in the Attributes manager and then enabling the Reflection option.

Now Reflection is listed as a tab in the Attributes manager, which means that we can animate the reflection parameters as well. At the moment the reflection is set to 0%.

► In the Attributes manager, click the Color tab.

► Shift-click on the words Color and Brightness to multi-select the two parameters.

► Right-click and select Animation > Add Keyframe from the context menu.

This records the color and brightness information for the material at frame 0. Note that as soon as we record keyframes, a red dot appears. This red dot tells us that there is a keyframe for this parameter at this frame. If we move the time slider, the red dot changes to a red circle. A red circle means that this parameter has a keyframe at some frame or frames, but not at the current frame.

Also note that, when animating a material, the color RGB sliders are independent of the Brightness slider and can be animated separately.

► Click the Reflection tab.
Command-click on a Mac if you have only one mouse button.

Right-click on Brightness and select Animation > Add Keyframe from the context menu.

To animate the reflection we first have to record a keyframe with 0% reflection brightness at frame 0.

Click the Specular tab.

Shift click on Width and Height to multi-select these parameters.

Right-click and select Animation > Add Keyframe from the context menu.

The specularity is very low at the start of the animation, suitable for clay. We have now recorded keyframes for all the parameters we need to change to morph this material into shiny metal.

Drag the time slider to the end frame of the animation.

In the Attributes manager, click the Color tab and change the color for the clay material to white (R=100%, G=100%, B=100%) and set the Brightness to 0%.

This results in black. We could simply set the brightness to 0% without changing the color sliders, which would also give us black, but setting the color sliders to white will result in the red color fading to grey and getting darker at the same time.
Shift-click on Color and Brightness to multi-select these two parameters.

Right-click and select Animation > Add Keyframe from the context menu.

Click the Reflection tab and adjust the Brightness slider to 100%.

This makes the material 100% reflective.

Command-click on a Mac if you have only one mouse button.

Right-click on Brightness and select Animation > Add Keyframe from the context menu.

Record a keyframe for 100% reflectivity.

Click the Specular tab

Set Width to 50% and Height to 100%.

This makes the specular highlight sharper and much brighter than at the beginning of the animation.

Shift click Width and Height to multi-select these two parameters.
Command-click on a Mac if you have only one mouse button.

Record a keyframe for the new specularity values.

Right-click and select Animation > Add Keyframe from the context menu.

From the main menu, choose Render > Render Settings and, on the Save page, define a path and a filename for the QuickTime movie we are about to render.

Choose a location which is easy to find, e.g. the desktop folder.

Select Render > Render to Picture Viewer to create a QuickTime movie of the animation.

These are a few frames of the rendered animation.

The rendering will take a few minutes. When finished, double click the QuickTime movie and watch the clay ball morph into a puddle of liquid metal. When played backwards, this effect looks even better. Just imagine a character instead of the red ball morphing out of the liquid metal!

Summary

Working through this tutorial has probably given us many ideas for other effects and parameters that could be animated. We could animate the brightness and the cone angle of a spot light, or the waviness of a water surface by animating the parameters of the water shader. To visualize the aging of a metal construction, we could animate a material going from shiny metal to rusty and old by animating the diffusion and bump values. Almost every parameter in CINEMA 4D can be animated like this, the possibilities really are endless.
Set Driven Keys 1

Use one parameter to directly control another by using the Set Driven Keys. These will reduce the amount of manual work you need to do.

Set Driven Keys is a powerful and flexible way to animate objects where one parameter (the driver) can control one or more other parameters (the driven). Set Driven Keys can be used in any situation where an alteration of one object will have a relationship to another. A dimmer switch controlling a light bulb brightness, a button controlling a camera shutter, even a spinning cog causing another to rotate can be set up with the Set Driven Keys feature.

This tutorial is an easy example so that we can become familiar with the concept. (A more advanced example follows in Set Driven Keys 2.) In this small example, all we are going to do is control the radius of a sphere with the movement of a cube.

**Load the scene named ‘SetDriven Simple Before.c4d’ from the Animation - SetDrivenKeys 1 folder in the CINEMA 4D Tutorials folder.**
We want to control the sphere’s radius by moving the cube.

- In the Object manager, select the Cube.
- In the Attributes manager, click the Coord. tab.

The Coordinates page shows the current position (P), scale (S) and rotation (R) values of the cube.

Command-click on a Mac if you have only one mouse button.

- Right-click on the letter Y of P.Y and select Animation > Set Driver from the context menu.

This defines the cube’s Y position as the driver parameter.

The cube’s Y position is now the driver, it is the controlling parameter. Note that there is a difference between clicking on the Y and the P in P.Y. Clicking on the P means that all three position parameters get selected while clicking on Y selects only the Y position.

- In the Object manager, select the Sphere.
- In the Attributes manager, click the Object tab.
Command-click on a Mac if you have only one mouse button.

Right-click on the word Radius and select Animation > Set Driven (Absolute) from the context menu.

This makes the sphere’s radius the driven parameter.

There are two different commands for Set Driven. Absolute means that the value of the first parameter is directly used as the value for the driven parameter. In our case the Y position of 100 results in a radius of 100. Relative, on the other hand, changes the parameters in relation to their current values. This doesn’t make a difference in our case but it would in others. Imagine controlling the position of one object with another. Using the absolute mode, object 2 would jump to exactly the same position as object 1, but using the relative mode object 2 would remain at the same position and move only relatively to the movement of object 1.

Select the Move tool and in the viewport drag the cube up and down to see how this affects the sphere.

The radius of the sphere is now controlled by the cube’s Y position. This is all there is to do if you want to control one parameter with another parameter.

**Summary**

This a very powerful feature of CINEMA 4D. Here we have seen an extremely simple example of how one object’s properties can affect another object. The uses of this feature are endless and we’ll look at another example in the next section.
Set Driven Keys 2

Here’s another fascinating example of connecting object properties so that the animation of one controls the animation of another.

This example of Set Driven Keys is a bit more complex than the previous one and shows us a real practical use for this feature. We will set up a bone chain in a scorpion tail so that we need only to rotate the first bone to make the whole bone chain curl. We will see from this project that Set Driven Keys are not limited to only two interacting parameters — one parameter can control any number of other parameters.

- Load the scene named ‘ScorpionTail Before.c4d’ from the Animation - SetDrivenKeys 2 folder in the CINEMA 4D Tutorials folder.

The scorpion tail already has a bone chain assigned to it and when we rotate or move any of the bones we can deform the tail. What we want to do now is make the bones curl up automatically.
In the Object manager, select Bone1.

This will be the controller bone. This is the first bone of the bone chain — the parent bone. We want to use this bone to control the rotation of its child bones.

In the Attributes manager, click the Coord. tab.

On the Coord. tab you can see the bone’s current position, rotation and scale values. We are interested only in the rotation (R) values.

Command-click on a Mac if you have only one mouse button. Right-click on the letter R (any of the 3) and select Animation > Set Driver from the context menu.

Right-clicking on the letter R affects all three rotation axes.

We want to use all three rotation axes as the driver, not just a single axis, so instead of right-clicking on H, P or B we right-click on the R next to one of these values.

In the Object manager, select Bone2.
In the Attributes manager, right-click on the letter R (any of the 3) and select Animation > Set Driven (Absolute) from the context menu.

This defines the rotation of Bone2 as the driven parameter.

The rotation of Bone2 is now controlled by the rotation of Bone1.

In the Object manager, select Bone3.

In the Attributes manager, right-click on the letter R (any of the 3) and select Animation > Set Driven (Absolute) from the context menu.

The rotation of Bone3 is now also controlled by Bone1. It is not necessary to select Set Driver for Bone1 again, CINEMA 4D remembers that we did this earlier. Once we have chosen Set Driver for a parameter, we can choose Set Driven for as many other parameters as we like.

Repeat the procedure from the previous step for all the remaining child bones.

The rotation of all child bones is now controlled by Bone1.
In the Object manager, Select Bone1.

Select the Rotate tool and in the viewport rotate Bone1 around the red X axis.

Rotating Bone1 results in the tail curling up.

When we rotate Bone1 the whole bone chain curls up. (Drag the red object axis for the best effect.) Now it is very easy to animate the scorpion tail stinging.

**Summary**

This example shows only a fraction of what we can do with Set Driven Keys. We could control the brightness of an array of lights by moving an object up and down, simulating a dimmer switch. Another example might be a muscle bulging where the rotation of the lower arm bone controls the scaling of a muscle bone in the upper arm.

Set Driven Keys is actually a simplified version of the XPresso node system. Whenever we use Set Driven Keys, a small XPresso setup is created behind the scenes. Open the XPresso nodes window by clicking on one of the XPresso tags and see how these relationships have been set up. The next tutorial shows how to work with XPresso to create more complex relationships.
XPresso 1
Clamp and User Data

As a gentle introduction to the powerful world of XPresso we will create a simple expression which prevents an object from falling below a certain height.

Sometimes there’s a need to limit an object’s range of motion. A typical example would be a foot that must not penetrate the floor. These kinds of situations are handled with the use of Clamp nodes in the XPresso node editor.

Adding user data enables us to control the XPresso set-up from within the Attributes manager, providing us with maximum flexibility while at the same time maintaining our XPresso settings.

In this tutorial we will prevent a cube from sinking below a certain Y position.

► Create a new scene by selecting File > New.

► Select Primitive > Cube from the Objects menu.

The cube’s parameters don’t need to be modified.

► In the Object manager, right-click on the Cube object and add an XPresso expression to it by selecting New Expression > XPresso Expression from the context menu.

When an XPresso expression tag is added to an object, the XPresso Editor opens.
Drag the Cube from the Object manager and drop it into the XPresso Editor.

By adding an XPresso expression to the cube and dragging it into the XPresso Editor, we create an object node for the cube.

When we drag the cube into the XPresso Editor an object node is created that contains all the data necessary to integrate the cube with other XPresso items.

Click on the red square on the Cube node and create an output port for the global Y position from the Coordinates sub-menu, resizing the Cube node afterwards as required.

Since the position of the cube has to be observed on the y-axis, the corresponding output port of the cube object node has to be present. Note that a differentiation between global and local data has to be made when using these ports; using global values makes sense if an object is positioned within a hierarchy but its parent objects are not to be taken into account.
Command-click on a Mac if you have only one mouse button.

In the XPresso Editor, right-click in an empty area of the window and select a Clamp node from the context menu (New Node > XPresso > Calculate > Clamp).

Link the Cube node’s Global Position. Y output port to the Clamp node’s Value input port by clicking on the output port and dragging a connection to the input port.

The Clamp node checks whether input values fall within defined limits.

The Clamp node is going to be used to check that the Position. Y value falls within certain limits. If it is outside those limits, a value closest to the limit will be output; if the input value falls within the limits, it will be passed on to the output port.

Select the Clamp node by clicking on its title bar and in the Attributes manager, on the Parameter page, set a very high value for Max, 1000000 for example.

The upper limit of the allowed range for the Clamp node is set so high that practically no clamping at all is present at the upper end.

By using the Attributes manager, values can be preassigned to input ports without the need for an extra node such as the Constant node. Since a very high value is set for the Max input port, a clamp towards the top is virtually disabled, i.e. we can move the cube as far as we like up the Y axis. The Max value represents the maximum allowed value for the particular clamp.
In the Object manager, select the cube and, in the Attributes manager, add a user data field by selecting User Data > Add User Data from the Attributes manager menu.

The Attributes manager enables us to assign data to objects that can then be accessed within the XPresso Editor.

The upper limit of motion for the cube has already been defined through the Clamp node’s Max value. We want the lower limit of the Y position to remain variable (for later modification outside the XPresso set-up) and therefore we should not set a value for the Min port. This port is necessary to define the limit but, unlike the actual Y position of the cube, it cannot be accessed easily from outside the XPresso set-up. That’s what user data is for — values that can be individually manipulated and sent to object nodes in the XPresso set-up.

Note that we should make sure the cube is selected and not the XPresso tag behind the cube, otherwise the user data will be assigned to the XPresso tag and not the cube object node.

In the Add User Data dialog, type in the name Border.

When we add user data, a dialog opens automatically that enables us to define all the necessary settings. First, we choose an appropriate name. The name Border seems to fit as this value will control the lower boundary of the cube’s freedom of movement.

Set Data Type to Float.

As the user data will control the lower limit of the Clamp node, the data type has to be the same as that of the clamp. So the user data has to be of the type Float — numbers with decimal places, that is, as opposed to whole numbers.

Set Interface to Float.

Setting the Interface to Float gives us an input box on the User Data page in the Attributes manager into which we can type a floating point number, which will be used to set the lower limit of the clamp.
Set Unit to Meter.

Set Min to -1000 and Max to +1000.

The Min and Max values define the range of values for which the user data operates. Values outside them will not be accepted.

Set Step to 1 and click OK to close the dialog.

Lastly, we set a step size for the interval. A step size of 1 will increase or decrease the value within the Min/Max boundaries by 1 unit.

When creating user data, a dialog will open that enables us to set the name, data type, and boundaries of the value range.

Note that we can add any number of user data fields to an object, as we see fit. In our example, a single user data field is enough.

Click on the Cube node’s red square and select User Data > Border from the output port list.

The Border user data can be selected from the cube object node’s output port menu.

If user data has been assigned to an object, it can also be used as a normal port in the node’s output port list; it will output the value that is set in the Attributes manager. In this, Border is 0 (the default value).
In the Cube node, click on the Border port name and drag it up so that it is above the Global Position.Y port.

Connect the Cube node’s Border output port to the Clamp node’s Min input port.

The connection between the Cube node’s Border port and the Clamp node’s Min port enables the Border user data value to control the lower limit for the Clamp node.

The connection between the Border port and the Min port continuously sends the user data Border value to the Min port of the Clamp node. This means that, even without the XPresso Editor being open, we can make constant changes to the lower limit of the cube’s range of motion along the Y axis from within the Attributes manager. The only requirement is that the cube has to be selected in the Object Manager, otherwise, the user data of the cube will not be displayed in the Attributes manager.

Command-click on a Mac if you have only one mouse button.

Note that if we don’t like the Float input box, we can always change it to a slider by right-clicking on its name in the Attributes manager selecting Edit Entry from the context menu and then changing Interface to Float Slider.

In the Object manager, drag the cube and drop it into the XPresso Editor once again.

The clamped position of the cube is output as values from the Clamp node. To have an effect, this value has to be rewritten into the Y position of the cube. A new cube-node is needed for this, which is created by dragging the cube from the Object Manager into the XPresso Editor once again. This way, several object nodes of the same object can be used in the same set-up.
In the XPresso Editor, click on the blue square in the Cube node and select Coordinates > Global Position > Global Position.Y from the list of input ports, resizing the node afterwards as necessary.

Connect this Global Position.Y input port to the Clamp node’s output port.

Now, when we move the cube in the viewport, we’ll notice immediately that we cannot move it in the Y direction below the preset user data value Border. (By default this is 0; try changing it.)

**Summary**

We have seen that XPresso nodes can have input ports and output ports. A node’s input port receives values from another node’s output port. In the Attributes manager we can control what happens to a value after it has entered an input port. Adding user data enables us to send our own values to an input port.
XPresso 2
Controlling Materials and Tags

Imagine a ball bouncing along a path, and whenever the ball hits the path it changes color...

XPresso can be used to combine parameters from different objects, tags and materials. In this tutorial we will be dealing with a combination of three elements. We are going to make a ball bounce up and down while moving along a spline path, and if the ball touches the spline path we will change the color of the ball.

► Create a new scene by selecting File > New.
► Select Primitive > Sphere from the Objects menu.

The sphere is our bouncing ball.

► Select Spline Primitive > Circle from the Objects menu.

We will be using the circle spline as the path along which our ball will bounce.
With the Circle spline selected in the Object manager, in the Attributes manager, on the Object page, ensure the Plane option is set to XZ.

The ball will be bouncing along this spline.

The ball will be bouncing along this spline.

We have put the spline on the XZ plane so that it is flat on the ground. The sphere will be jumping up and down along the global Y axis. We don’t need to make any other changes to the objects.

Command-click on a Mac if you have only one mouse button.

An Align to Spline expression automates the Sphere’s movement along the spline.

Using the Align To Spline expression, an object can be made to follow the path of a spline. To tell an object which spline path to follow, the spline needs to be dragged and dropped into the Spline Path input box of the object’s expression tag. By dragging and dropping the Circle into this input box we are telling the Sphere to follow the path of the Circle spline.
The Align To Spline options in the Attributes manager also include a Position value, displayed as a percentage. This value controls the Sphere’s position on the spline; changing it makes the Sphere move on the spline. To make the Sphere move along the entire length of the spline, this Position value needs to be animated from 0% to 100%.

- **In the Object manager, right-click the Sphere and select New Expression > XPresso Expression from the context menu.**

When adding an XPresso expression to an object, the XPresso Editor window opens automatically.

- **Drag the Align To Spline tag from the Object manager and drop it into the XPresso Editor, resizing the node afterwards if desired.**

If a tag is dropped in the XPresso window, a node is created for the tag.

- **In the XPresso Editor, click on the blue corner of the Align To Spline node and select Tag Properties > Position from the menu.**

Clicking on the blue corner of the node lists all of the tag’s parameters that can be used as input ports. We are adding the Position port to be able to control with XPresso the Sphere’s position on the spline. A Position value of 0% corresponds to the starting point of the spline, while 100% places the Sphere at the end. (For a circle spline, by default this start/end point is where the spline intersects with the positive X axis.)
Command-click on a Mac if you have only one mouse button.

In an empty area of the XPresso Editor, right-click and select New Node > XPresso > General > Time from the context menu.

Drag a connection between the Time node’s Real output port and the Align To Spline node’s Position input port.

Using the Time node, a parameter can be controlled during the animation playback.

The Time node can be used to output the time of the running animation. The Real port outputs the animation time in seconds, with a Real value of 1.0 corresponding to a percentage value of 100% (we’ll adjust the time in a moment).

Since we have connected the Real port to the Position port (which is a percentage) the Position will change from 0% to 100% over one second of animation. So, in our case, the Sphere will move from the start to the end of the spline in exactly one second. We can test this if we press the Play button in the animation toolbar.
If the animation lasts longer than a second, the sphere will not continue moving on the circle spline because position values beyond 100% are automatically clamped to 100%. This also means that we don’t have to worry about manually clamping values input in the position port.

- **Select New Node > XPresso > Calculate > Math from the context menu in the XPresso Editor.**

We have added a Math node to control the speed of the Sphere on the spline.

- **In the Attributes manager, set Function to Divide.**

- **Change the Input [2] value to 3.**

We have set it to a division by 3 because we want to divide the Real value of the Time node by 3 to slow down the Sphere. It will then go around the spline in three seconds instead of one second. If we wanted the Sphere to go around the spline in two seconds, we would have to type a value of 2 into the lower input port of the Math node.

- **In the XPresso Editor, connect the Math node’s upper Input port to the Time node’s Real port.**
Connect the Math node’s Output port to the Align to Spline node’s Position port.

The Math node divides the time value by 3, making the Sphere move along the spline at only a third of the original speed.

The old connection between the Time node and the Align To Spline node is deleted as soon as the Math node is connected to the input port of the Align To Spline node. That’s because an input port is allowed only one connection.

In the XPresso Editor, select New Node > XPresso > Calculate > Formula from the context menu.

We will use this Formula node to control the up and down movement (the bouncing) of the Sphere.

Create an input port by clicking on the blue corner of the node and selecting Value from the menu.

In the Attributes manager, under Node Properties, type ABS(SIN($1*360)) into the Formula input box.

Sine waves are perfect for creating bouncing motions. A standard sine wave, however, offers values only between -1 and +1. Besides that, the Sphere shouldn’t slow down while approaching the ground. We need a curve which abruptly changes from a down movement to an up movement.
The solution to this problem is to flip the negative values of the sine wave into positive ones. We can use the ABS function for this, which converts all negative values to positive ones.

The negative values of the sine wave (left) can be converted to positive ones (right) by using the ABS function.

The ‘$’ symbol is followed by the number of the input port of the Formula node, and this enables us to include an external value in the formula. Since the Formula node can work with degree angles, without the need of a prior conversion from Radians, the formula $\text{SIN}(\$1*360)$ creates an animated sinus oscillation. The ‘$1$’ variable inputs the time from the Time node, which results in a full sine wave per second of the animation.

- In the Xpresso Editor, connect the Time node’s Real port to the Formula node’s input port (blue circle).

- In the Object manager, select the Sphere and in the Attributes manager select Add User Data from the User Data menu.

- In the Add User Data dialog, type Height into the Name box.

- Set Unit to Meter and click OK to close the dialog.

The user data will be used to control the bouncing height of the sphere.

The sine wave generates values between -1 and +1 which are converted to values between 0 and 1 using the ABS function. However, we want to be able to make the bouncing height smaller or bigger, so we need to be able to scale the waveform. To achieve this we have added User Data, displayed as a new input box on the User Data page in the Attributes manager.
► In the Object manager, drag the Sphere and drop it into the XPresso Editor, resizing things afterwards as required.

► Click the red corner of the Sphere node and select User Data > Height from the menu.

As soon as user data is created in the Attributes manager, it is also accessible in XPresso. It is listed together with all other parameters. We have now created an output port for the Height user data which we added to the Sphere in the previous step. We are now able to connect our Height user data port with other ports and use it for calculations.

► Right-click in the XPresso Editor and select New Node > XPresso > Calculate > Math from the context menu.

► In the Attributes manager, under Node Properties, set Function to Multiply.

► In the XPresso Editor, draw a connection between this Math node’s first input port and the Sphere node’s Height output port, reorganizing the contents of the editor window afterwards as required.

► Draw another connection between the Math node’s second input port and the Formula node’s output port.

We have now connected both the Height user data and the Formula node with a Math node which multiplies both outputs. The Height is multiplied by the outcome of the Formula. The waveform can now be scaled in the Attributes manager using the Height input box for the Sphere; it is no longer limited to values between 0 and 1. By multiplying the Formula by our own new Height value we can control the bouncing height of the sphere interactively in the Attributes manager... but not until we have actually connected the sphere with the Math node.

► Drag the Sphere from the Object manager into the XPresso Editor once more.

► In the XPresso Editor, create an input port by clicking on the blue corner of the new Sphere node and selecting Coordinates > Global Position > Global Position.Y, resizing the node afterwards as required.
Connect the Global Position.Y input port to the Math.Multiply node’s output port.

The Y position of the Sphere is controlled by the formula and can be interactively adjusted with the Height parameter.

In the Object manager, select the Sphere’s XPresso tag.

In the Attributes manager, set Priority to Dynamics.

The Y position of the Sphere (the bounce height) is now controlled by the formula.

The priority of the expression needs to be adjusted.

The Sphere doesn’t bounce yet. That is because expressions are always evaluated from left to right. Where expressions are positioned in the hierarchy and in which order they are placed on the objects is important. If several expressions are applied to an object, one may override the other. In our case the up and down bouncing along the Y axis is being overridden by the Align to Spline expression which is also assigned to the Sphere and positioned to the right of the XPresso expression in the Object manager.

In this case there are two possible solutions. We have chosen to set the priority of the XPresso expression to Dynamics so that it is calculated after all other expressions. However, we could have dragged the XPresso tag to the right of the Align to Spline tag, which would have had the same result of the sphere continuing to move along the spline but the Y position now being controlled by our XPresso expression. So the Align to Spline expression is controlling only the X and Z axes and no longer the Y axis.

In the Material manager, create a new material by selecting File > New Material and name it Color.

Drag the material from the Material manager and drop it onto the Sphere in the Object manager.
Drag the material from the Material manager and drop it into the XPresso Editor.

The material will be controlled with XPresso to make the sphere change color when it hits the ground.

Materials can be dragged into the XPresso Editor to create a node which includes all of the material’s parameters that can be used as inputs or outputs. Many of the parameters, such as color and brightness, can be easily controlled using numbers or vectors.

Create input ports for the G and B (green and blue) components of the material color by clicking on the blue corner of the new node and selecting Color > Color > Color.G and Color > Color > Color.B, afterwards adjusting the size of the nodes as required.

We want to control the green and blue color components to make the sphere change from white (RGB all at 100%) to red (R at 100%, GB at 0%). The R component doesn’t need to be controlled by XPresso, it will always remain at 100%.

Right-click in the XPresso Editor and select New Node > XPresso > Calculate > Clamp from the context menu.

In the Attributes manager, set Max to 1.
In the XPresso Editor, connect the Clamp node’s output port to both the Color.G and the Color.B input ports of the Color node.

The Clamp node ensures that only values between 0 and 1 are fed to the color input ports. 0 corresponds to 0% and 1 to 100%.

Connect the Math.Multiply node’s output port of the to the Clamp node’s Value input port.

Using the Clamp node we ensure that only values between 0 and 1 are fed into the color input ports. Any values that are output by the Math node and are bigger than 1 get reduced down to 1 by the Clamp node.

The Math node (which is connected to the formula) outputs values between the lowest point in the up and down movement of the sphere and whatever value is typed into the Height user data input box. Since the Height will probably be much higher than 1 most of the time (otherwise the sphere would hardly bounce at all) we are using the Clamp node to ensure the sphere changes color only when it is very near the lowest point. The material will change from white to red when the sphere is at a position between 0 and 1. This creates a very quick color change when the sphere hits the ground.

In the animation toolbar, press Play to watch the result.

Summary

We have seen how the parameters of objects, tags and materials can all be used in XPresso expressions and how the value of one or more parameters can be used to change the value of one or more other parameters. Select the Sphere in the Object manager to play with the Height value on the User Data page in the Attributes manager; try setting Height to 100 to start with.
XPresso 3
Vector Manipulations

Vectors are one of the most used elements in 3D space. They define the position, orientation and even the size of an object.

Before working through this XPresso tutorial, please ensure that you have completed the previous two. In this tutorial we will assume a certain familiarity with XPresso.

A vector is made up of three values, such as (-1.5;9;13). The numbers stand for the XYZ (position) or the HPB (orientation) components of a vector. The manipulation of the orientation and lengths of vectors can be used for a variety of applications; in this tutorial we are going to connect a virtual leash between two objects.

Imagine walking a dog. The dog is allowed to move about in the radius defined by the leash. As soon as we move, once the leash tightens, the dog has to follow. To extend this simple example we’ll make the length of the leash variable, like an automatic roll-up leash.
Create a new scene by selecting File > New.

Select Primitive > Sphere from the Objects menu and name it Dog.

Select Null Object from the Objects menu and name it Master.

The new scene contains a sphere and a null object.

In the Object manager, select the Master object and add an XPresso tag to it by selecting New Expression > XPresso Expression from the context menu.

Drag the Dog and Master objects from the Object manager and drop them into the XPresso Editor.

Dog and Master will be used as object nodes in the XPresso set-up.

To use the position of Dog and Master in the calculations, both objects must be placed as object nodes in the XPresso set-up.
In the XPresso Editor, for both object nodes select the Coordinates > Global Position > Global Position output port from the port list on the red square in the node title bar.

The global positions of Master and Dog are selected from the output port list of both object nodes.

In order to calculate the distance between two objects, the global positions should be used, even if the objects are placed within different hierarchies. In this case, the objects are on the same hierarchic level. Using global values means the objects can be moved to new hierarchies at a later point without affecting the way the XPresso expression works.

Create a new node by selecting New Node > XPresso > Calculate > Distance and connect its input ports to the Global Position ports of Master and Dog.

The distance node calculates the distance between the position of the master and the dog.

The Distance node calculates the distance between two given positions. When we connect the Distance node’s first input port to the Dog node’s Global Position port and do the same for the Distance node’s other input port and the Master node, the Distance node’s output port will send the distance between Master and Dog to whatever else is connected to it.

Note that the positions of Dog and Master are vectors, but the distance calculated by the Distance node is a number.

In the Object manager, select the Master object and add user data to it by selecting User Data > Add User Data from the Attributes manager menu.

Since the leash needs to be controlled by the master, the Master object needs to contain the user data entry. As we’ve seen in the previous example, user data entries have the benefit of giving us control of the XPresso values from outside the XPresso set-up. In this case, the distance user data will be used to modify the length of the leash.

In the Add User Data dialog, type Distance into the Name input box.
Before adjusting the actual user data value for Distance, the current value is
0. Here we are saying that, when adjusting the actual user data value for Distance, it
cannot be shorter than 0 m nor longer than 1000 m.

Set Step to 1 and click OK to close the dialog.

In the Object manager, select the Master object and in the Attributes manager,
on the User Data page, set Distance to 200.

For the purposes of this example, here we have set the length of the leash to 200 m.

In the XPresso Editor, select the User Data > Distance port from the Master
node’s output list (red square).

In the XPresso Editor, create a Clamp node from the context menu (New Node >
Xpresso > Calculate > Clamp).

Connect the Distance node’s output port to the Clamp node’s Value input port.

Connect the Master node’s Distance port to the Clamp node’s Max port.
In the Attribute Manager, set the Clamp node’s Min input port to 0.

Since the upper limit of the clamp’s value range is defined by the Distance user data value, the result sent from the Clamp node’s output port cannot exceed the Distance user data value.

The actual distance between dog and master, which is evaluated by the Distance node and input into the Clamp node, only gets as far as the clamped output when its value lies within the boundaries of zero and distance. Since the distance between two objects cannot sensibly be less than zero, we choose this value as the Min port of the Clamp node. This value is a constant and could be set within the Attributes manager directly.

In the XPresso Editor, select a Math node from the context menu (New Node > XPresso > Calculate > Math).

In the Attributes manager, set Data Type to Vector and Function to Subtract.

In the XPresso Editor, connect the Dog’s Global Position output port to the Math node’s top-most input port.

Connect the Master node’s Global Position output port to the Math node’s lower input port.

The Math node enables one vector to be subtracted from another in order to calculate the orientation vector between two positions.

The Math node can be used for a great variety of calculations. It offers the basic functions — add, subtract, multiply, divide and modulo — which can be chosen from the Attributes manager when the Math node is active. The Math node is not limited to floating-point values, it can also deal with vectors.
Note that both input ports have to receive values of the same data type, otherwise values are matched automatically and this might cause floating-point digits to disappear or vectors to be converted into simple scalar values, delivering unwanted results. In our case, a position vector is being subtracted from another position vector, delivering a vector again. This resulting vector describes the orientation which one position has to take in order to reach the other position.

The order of the position vectors in the calculation is significant. Generally, one could say that subtracting the position of the current location from the position of the target location delivers the orientation vector that shows the path from the current location to the target location. In the case of our dog and master example, the Math node calculates the orientation which is needed by the master to reach the dog. The vector calculated by the Math node also contains information about the distance between dog and master in its length. Since the clamp has not been applied to the dog object, this value may fall outside the allowed leash range. We don’t need to worry about this because the proper distance between master and dog is calculated by the Clamp node.

- In the XPresso Editor, select a Universal node from the context menu (New Node > XPresso > Adapter > Universal).

- In the Attributes manager, set Data Type to Normal.

- In the XPresso Editor, connect the Math node’s output port to the Universal node’s input port.

Two different types of data are available now that have to be combined. The orientation vector between dog and master has been calculated by the Math node; it describes which way we have to go to reach the dog from the position of the master. The length of the vector may lie outside the boundary set up by the Distance user data entry. The other value is the distance between dog and master. This value has been changed to a valid number by using the Clamp node and Distance value.
The task is to change the Math node’s orientation vector to a length calculated by the Clamp node. It would be useful to get the resulting vector of the Math node to have a length of 1. Afterwards it could simply be multiplied by the proper distance given by the Clamp node.

Fortunately, vectors of length 1 are common mathematical elements. They are called normal vectors, or normals. You may be familiar with these type of vectors from dealing with polygons — the small yellow lines displayed when selecting a polygon are normals. The conversion of a vector into a normal, which still points in the same direction but has a unified length, is handled by the Universal node. This ‘adapter’ node facilitates the conversion of one data type into another, a vector into the Normal data type for example.

- **In the XPresso Editor, select a FloatMath node from the context menu** (New Node > XPresso > Calculate > FloatMath).
- **In the Attributes manager, set Data Type to Vector and Function to Multiply.**
- **In the XPresso Editor, connect the Universal node's output port to the FloatMath node’s input-port.**
- **Connect the Clamp node’s output port to the FloatMath node’s Float Value port.**

As described earlier, the Math node converts different data types automatically, to conform to a unified data type. The FloatMath node has almost the same functionality as the Math node except that it can work with different data types without having to convert them. This way, the normal vector of the Universal node can be multiplied with the real value of the Clamp node output effortlessly. The result is an orientation vector of valid length between dog and master.

- **In the XPresso Editor, select another Math node from the context menu.**
- **In the Attributes manager, set Data Type to Vector and enable the Add function.**
In the XPresso Editor, connect the Master node’s Global Position output port to the new Math node’s input port.

Connect the FloatMath node’s output port to the new Math node’s other input port.

Since the calculated orientation vector needs an origin in order to point to a target, the position of Master has to be added to the FloatMath node’s orientation vector.

The orientation vector calculated by the FloatMath node points in the right direction and has the proper length, but it does not provide an origin. Since the orientation vector defines the path between master and dog, this does not present a problem. We simply use the position of master as the origin of the orientation vector pointing at dog.

This combination of position and orientation vector can be handled with a small addition in the Math node. Because both values are of the same data type (vector) the Math node will not have to convert data types. As we are using an Add function with the Vector data type in the Math node, the order in which the values are connected to the inputs is not important.

In the Object manager, drag the Dog object and drop it into the XPresso Editor to create another Dog object node.

So far, the XPresso set-up calculates a new position for the Dog object if the length of the leash, which is controlled by the Master object’s user data Distance, does not exceed the current position. To complete the set-up, the new global position has to be rerouted back into the Dog object.
In the XPresso Editor, click on the blue square in the new Dog node and select a Global Position input port from the list.

Connect the new Dog node's Global Position input port to the output port of the last Math node we created.

The newly created global position has to be transferred on to the Dog object by a Dog object node.

This connection ensures that the Dog object cannot move farther away from the master than the Master's user data Distance allows (which we earlier set to 200 m). On the other hand, the master can move about freely, automatically dragging the dog along as soon as the leash tightens.

Summary

We have seen how a complex relationship can be built up between two objects so that no matter how we independently animate one object, the motion of a related object will be limited by a strict set of rules.
Rendering Still Images

Rendering is a process which is required for all of the images you will create. Read on to learn how these rendered images are created.

Rendering is what we do when we want to see our 3D objects in a higher quality than the viewport can display. The viewport cannot show the full details of an object, such as reflections or light bending refraction, because these require time to calculate. The rendered images are rendered out to a file in whichever file format and at whatever pixel resolution we require. We can either do a preview rendering of our scene directly into the viewport (as we have done several times already in previous tutorials), or we can choose to render it to the Picture Viewer, which will give us the extra option of saving the picture to a file.

The time required for rendering an image can vary greatly. Simple scenes, such as those used in this tutorial manual, will take between a few seconds and a few minutes to calculate. Larger files, such as detailed architectural scenes, may possibly take several hours, depending on the hardware used.

- Load the file named ‘Rendering1.c4d’ from the CINEMA 4D CD.

This is the file we will be using to illustrate the basics of rendering. The settings in this file are all set to the defaults, the objects are there only to better show the rendering process.
Click once on the Render Settings icon on the top toolbar.

Open the render settings dialog by clicking on its icon.

Because of the nature of 3D rendering, all settings must be entered before we save the file. Traditional image packages, such as Photoshop and Paint Shop Pro, will save the images almost instantly, but because 3D images can take a while to generate it is best to set all of the resolutions and file formats first. CINEMA 4D will automatically display the results and save the file when rendering is complete.

On the left-hand side of the Render Settings dialog, click the word Output to display the Output page.

The different properties of the rendering are listed on the left.

The Output page is where we specify the output size of the image or animation we are creating. If we are intending to print our work we need to set the resolution much higher than the default; this may be several thousand pixels wide for 300 dpi output.

Set Resolution to 1024x768.

We are assuming you have a screen resolution of at least 1024x768.

The default settings are designed for quick, low resolution previews of our work so we need to increase the output size to a higher resolution in order to see any detail.

Go to the Save page (click on the word Save on the left).

The Save page enables us to choose how and where files are saved, including filenames and file formats.

Click the Path button, select a location and use the filename ‘Rendering1.tif’.

For this tutorial, pick an obvious location such as the desktop.

Once the image has been rendered, the picture will be saved to the location we specified and with the filename we chose.
Set Format to TIFF.

The TIFF format is one of the most widely accepted. The TIFF file format is one of the most widely used. TIFF files keep all of their original quality and allow an animation rendering to be paused between frames. If we use this format to render an animation we will need a suitable piece of software to join the images together afterwards.

Click OK to close the Render Settings dialog.

Select Render > Render to Picture Viewer from the main menu.

The Picture Viewer will hold the image currently being rendered.

Once these basic steps are set up we can click the Render button to tell CINEMA 4D to begin generating the 3D image. As the image renders we can see it appear in the Picture Viewer line by line. Depending on your hardware, this will take anywhere from a few seconds to several minutes. Once rendered, the picture will be saved to the location we chose earlier. This image can then be loaded into any video or art package that supports the TIFF file format.

Summary

We’ve learnt the importance of setting the various render options and we’ve changed the rendered size of our image from the default value. Feel free to browse through the other default settings, changing them and re-rendering to see their effect. CINEMA 4D can render to a wide range of image formats making it a perfect companion to other creative products.
Rendering Animations

CINEMA 4D can output to a number of different movie formats including QuickTime, AVI and separate frames. Here, we’ll take a look at some of the issues involved in creating an animation.

Still images and animation renderings have a different set of rules. It is fine for still images to take several hours to render, whereas most animations must be able to render a frame within a few minutes. Still images may require a high resolution of several thousand pixels if they are used for print whereas animations typically remain under a thousand. This tutorial will go through the process of modifying the Render Settings to create an animation.

- **Load the file named ‘MakeAnim.c4d’ from the Rendering - Animation folder in the CINEMA 4D Tutorials folder.**

  This file contains an animation but is currently set up to render a single still image. We will be setting up the scene to render as an animation by changing several of the settings.

- **Click once on the Render Settings icon on the top toolbar.**

  Clicking on this icon will open the Render Settings dialog, and it is from here that we can adjust them to suit our needs. Note that the Render Settings in this scene are currently set up for rendering a single, high resolution image, they are not the default settings.
On the left-hand side of the Render Settings dialog, click the word Output to display the Output page.

The Output settings are quite important to animation as they enable us to select the frame rate and resolution from a set of presets for most animation formats and media.

Set Resolution to 720x576 D1 PAL.

The current settings are far too high for animation. While lots of detail may be needed for a still image that will be looked at for seconds or even minutes, a frame of animation will only ever be seen for a fraction of a second. This means we can drop the resolution to 720x576.

Note that although technically we should also change the pixel aspect ratio, we have decided to leave it set to square pixels as the distortion will be negligible.

Set Frame to All Frames.

Rather than just rendering a single image we want to render all of the frames in our project in order to create the animation. Note that we can choose to set a manual number of frames if we don’t need to render the entire sequence.

Set Field Rend to Odd Field First.

TVs make use of interlace fields to improve the frame rate.

When creating animation for TV it is highly recommended that you enable Odd Fields First. Field rendering takes advantage of the fact that PAL televisions will display alternate lines of the image 50 times per second (60 with NTSC) to create a smoother animation.

Set the Frame Rate to 25.

European television sets use 25 frames per second.

PAL televisions use a full frame rate of 25 images per second (30 for NTSC).
Go to the Save page (click on the word Save on the left).

Just as with still images, all render settings need to be made before the rendering process proceeds. Doing this also stops pesky dialogs from asking for a filename and location with each test render we do.

Set Format to QuickTime Movie.

The Apple QuickTime player is available for most operating systems.

To keep things simple we will get CINEMA 4D to automatically compile the frames into a movie file so we don’t have to worry about joining them in a video editing package.

Click the Path button, select a location and filename of your choice.

As the file format is set to QuickTime, we need not worry about lots of image files being created, just the one movie file.

Go to the Antialiasing page.

Antialiasing is important for image quality.

Geometry antialiasing will give great results in little render time.

Ensure that Antialiasing is set to Geometry

When dealing with moving images, much of the detail will be lost to us, not only because of the reduced resolution, but also because each image is seen only for a fraction of a second. This means that the higher image quality of Best Antialiasing would barely make a difference to what we see, so there is no reason to slow down the rendering process by using it. However, in situations which involve large amounts of transparent or reflective objects, or many near horizontal or vertical lines, we should consider Best Antialiasing in order to avoid pixels flickering on and off.

Set Filter to Animation.

Filter out the jagged pixel edges.

Animations will benefit from selecting the Animation antialiasing filter. This will blur the image slightly more than the Still Image filter, but it helps to soften edges even more, reducing possible edge aliasing. For a still image the Animation filter would be too blurry, but for animations it can improve the perceived quality.

Click OK to close the Render Settings dialog.
Select Render > Render to Picture Viewer from the main menu.

As soon as we select this command, CINEMA 4D will begin creating the 3D images needed for the animation. Once these are all generated, it will compile them into a QuickTime movie. This part may take several minutes on slower machines. When the rendering has completed, we will find a QuickTime movie on our hard drive in the location we set earlier. Note that QuickTime must be installed to be able to play this movie. (The installer is available on the Goodies CD which came with CINEMA 4D.)

Summary

We have learnt several important things about how to render an image in CINEMA 4D. The image size, format and filename should all be selected before the rendering process begins. Rendering does not happen instantly. Animation consists of many still images connected together. The time it takes to render an image will vary depending on the project and the hardware used.

Rendering is an essential part of the production process, without it our models will remain merely a collection of numbers and calculations in the project file. We should always remember to budget in the rendering time when planning how long a project will take.
Multi-Pass Rendering

Here we will illustrate how to set up the multi-pass feature to create several layers in our images. These will help with post-processing.

Multi-Pass rendering is a tremendously powerful feature which should never be overlooked, especially by professional users with picky clients. Multi-Pass rendering is the ability to render certain objects or elements onto different layers or files with the intent to remix them later in post-editing. This may initially seem like more work for the same result, but take a moment to digest the following example. We’ve just finished rendering a huge 10,000 frame animation of an architectural fly through showing what it all looks like at midday. Perfect, but what would it look like at sunset or at midnight...?

This is a common request from a client. Using multi-pass rendering we can create all three scenarios in a single rendering with very little overhead. All we need do is set up the lighting and modeling for all three times of day and enable multi-pass rendering. This will then enable us to selectively add, remove or mix the different lighting set-ups after the rendering is complete. So there is no need to run off three separate renders, just display the light sources which make up our required scene.

In this tutorial we will illustrate how to go about setting up a scene for multi-pass rendering. Once this file is created we will then be able to load it into a 2D art package and continue making changes.
Load the file named ‘Multi-Pass.c4d’ from the Rendering - Multi-Pass folder in the CINEMA 4D Tutorials folder.

The file we've just loaded contains a complete scene with all objects and lighting set up and ready to go for a standard rendering.

**Click once on the Render Settings icon on the top toolbar.**

All of the various render settings, including multi-pass, are reached from the Render Settings dialog. The Render Settings dialog can also be opened from the Render menu at the top of the window.

**Go to the Multi-Pass page.**

The Multi-Pass page is where we will set up all of the various layers that we want to be created during the rendering process. This is also the place where we set which file format to use and to where the files should be saved.

**Set Separate Lights to All.**

The Separate Lights option decides whether or not separate layers should be created for each light source. The great advantage of doing this is that, in almost any 2D art package, individual light sources can be turned on or off after rendering or, even more usefully, tinted.

**Set Mode to 2 Channels.**

Each light source can create three separate channels for each component of the lighting, diffuse, specular and shadows. As we have chosen to create only two layers per light, the specular highlights and the regular diffuse lighting will be combined into a single channel, with the shadows being the second layer for each light.
> In the Channels drop-down menu, select Reflection, then Refraction, then Ambient.

Reflections, refractions and ambient lighting all feature in our scene. These are the three elements in the render which are not covered by the specific ‘per light’ channels. Reflections, refractions, luminous materials and environment maps all need to be added to the image to make it complete. Luminous materials and environment maps are combined in the Ambient channel as they use the same rendering method.

> In the Channels drop-down menu, select Object Buffer.

You can customize which layers will be generated. The default buffer ID will be fine. The Object Buffer enables us to create an alpha channel for specific objects. Each object in the scene can be assigned an ID, this ID is then rendered as a separate alpha layer in the multi-pass file.

> When prompted for a Group ID, Click OK to accept the default value.

The Object Buffer enables us to create an alpha channel for specific objects. Each object in the scene can be assigned an ID, this ID is then rendered as a separate alpha layer in the multi-pass file.

> Click the Path button and select a location and filename.

All render settings are made before the files are saved, this enables us to start a project rendering and not have to worry about entering a filename when it is ready to save.

> Go to the Save page and disable the Save Image option.

Disable the Save Image option to avoid saving an extra flat image with no layers. There are two types of render in CINEMA 4D, a regular render and a multi-pass render. As we are creating this file to show how multi-pass rendering can be used, we have disabled the regular render.
Go to the Output page.

The Output settings enable us to specify a pixel size. The Output options will affect both regular and multi-pass rendering. On the Output page we can decide which frames should be rendered at what resolution.

Set Resolution to 1024x768.

Changing the output size.

The default resolution of 320x240 is very low, increasing it to a more reasonable level will give a more detailed display of how light sources can be manipulated after rendering.

Click OK to close the Render Settings dialog.

In the Object manager, ensure the Interior object is selected then select New Tag > Compositing Tag from the File menu.

Compositing tags give us extra rendering options for each object.

Compositing tags can be applied to individual objects or whole groups; they provide greater control over render settings per object.

In the Attributes manager, on the Object Buffer page, enable Buffer 1.

Object buffers create extra alpha layers for certain objects.

A while back we added Object Buffer 1 to the multi-pass render settings, this will create an individual alpha layer for anything which is assigned to Object Buffer 1. Objects are assigned by setting the buffer ID in the render tag, as we have just done.

Select Render > Render to Picture Viewer from the main menu.

If everything is correctly set up in the render settings, CINEMA 4D will begin rendering an image and, upon completion, will save the layered file to the location we specified.
We may notice that the multi-pass render and the regular render give slightly varying results regarding color saturation and brightness. Although multi-pass rendering gives extremely good results, we should always keep in mind that 100% color accurate images cannot be achieved by combining and filtering the various layers.

► In the Picture Viewer, select Cold Light > Blend: Diffuse+Specular from the Channels menu.

You can see the lights appearing on the right-hand side of the car.

What we should see is the image as if only one light is shining and all other reflections and shadows are turned off. By reducing the opacity of this layer in a 2D art package we can effectively turn the light up and down.

Further editing of the multi-pass file will require a 2D art package that supports layered PSD files. For the remainder of this tutorial we will be using Adobe® Photoshop® to illustrate the various techniques for adjusting layers.
- **Start the 2D art application.**

  If we do not have sufficient memory to run both applications at once, we may have to save our project and close CINEMA 4D at this point.

- **Locate the multi-render file we saved from CINEMA 4D and load it into the 2D art application.**

  Photoshop is an ideal 2D art package.

  We should see each layer correctly composited into the scene, and each light will be in its own layer set. We may notice that the image doesn’t quite look the same as it did when it was displayed in CINEMA 4D’s Picture Viewer. This will be because the different render passes (layers) must be ‘added’ to correctly create the image. Sadly, Adobe™ Photoshop™ does not yet support an Add mode for mixing layers, so the Screen mode has been used instead; this results in slightly darker images.
Select the Warm Light Blend layer and adjust the color balance to give the image a red tint.

By tinting a single light source, it appears to make the light change color.

By adjusting the diffuse layer of a single light we can create the impression that the light is tinted, even though the original light source is pure white.

Select the Cold Light Blend layer and adjust the color balance to give the image a deep blue look.

Cooler, blue lights are often used for the background to focus the viewer’s attention elsewhere.

By changing the front light source to a rich red and the rear light source to a dark blue color, we can create a stylized sunset type lighting. If we compare this to the original image we can already see how it is becoming a much more interesting image.
Select the Warm Light Shadow layer and reduce its opacity to 50%

As this image was not rendered with radiosity, all of the shadows will be pure black, which is not natural. By reducing the opacity of shadows it will appear as if the warm sunlight is bouncing around the interior of the car to slightly illuminate the darker areas. The realism of the image should not be affected because there are still very dark shadows being cast by the colder light.

Summary

The final composition is now complete. After rendering a simple image, we have been able to adjust several components with ease. The changes we have made thus far would traditionally require the whole image to be re-rendered to test each change. We can see that, with a more complex scene, the multi-pass method will save us a tremendous amount of time.

We might recall adding a Compositing tag to the Interior object and enabling the Object Buffer for it. We should find that we have an extra Channel available in our 2D art package which contains an alpha image of the interior. This can be used to composite the car against any background of our choice.
Compositing Tags

In the multi-pass tutorial we assigned a compositing tag to an object. Here we illustrate how these can be used to increase productivity.

Compositing tags can be used to improve quality and reduce the render time of all scenes. If our scene looks good but there are one or two objects which show an antialiasing problem, we can single these out with a compositing tag. The compositing tag gives us extra options for various render settings on a per object basis, meaning we don’t have to turn up the quality for the entire scene.

- Load the file named ‘Compositing Tag.c4d’ from the Rendering folder in the CINEMA 4D Tutorials folder.

Here we have a scene file which already has the antialiasing set up to give acceptable quality and render times.

- Click the Render to Picture Viewer button and take note of the render time.

Although the results are fine, they could be improved.

If we look at the image we should see that the results aren’t too bad, although they could be improved. Similarly, the render time isn’t the worst ever seen but it could be faster. On a benchmark machine this scene takes 20 seconds to render.

- Select Render Settings from the Render menu.
Go to the Antialiasing page.

Antialiasing is so important that it has its own section in the Render Settings.

The Antialiasing settings provide us with control over all aspects of the way jagged edges are removed from the render. Currently, this page is set up to use a high level of antialiasing on complex areas.

Set both the minimum and maximum antialiasing levels to 1x1.

You can fine tune how much antialiasing should be used.

By setting the antialiasing levels to 1x1, the render time will be greatly reduced. We should note, though, that even with Best AA selected the edges of objects will always have the very high quality Geometry style antialiasing applied to them. The benefits of Best AA can be seen only on transparent, reflective or broadly textured objects.

Close the Render Settings dialog.

In the Object manager, ensure that the Glass object is selected and select New Tag > Compositing Tag from the File menu.

Tags give particular objects extra properties.

The Compositing tag will give extra render settings to this particular object. This enables us to single out problem areas and assign more processing time to them. The obvious benefit of this is that we don’t have to use high quality settings for the rest of the scene when it doesn’t really need them.
In the Attributes manager, enable the Force Antialiasing option.

By forcing certain antialiasing settings on this object we can ensure it will be rendered crisply and cleanly, without any jagged lines.

Click the Render in Picture Viewer button.

Once it has rendered we will notice that the rendering time has dropped — a benchmark machine renders the scene in 15 seconds rather than 20 (that’s 25% faster). This small optimization has not only given a better image quality, but it has also reduced the render time of the image by a significant amount. Although five seconds may not seem like much, if we were to render this image 50 times larger for print, or render 10,000 frames for an animation...

Summary

Graphics artists must have a certain technical knowledge of the tools they use, whether paint and canvas or mouse and monitor. With some basic understanding of the methods and techniques we can use to speed up our work we will be able to be far more productive without compromising quality.
Tips and Tricks

This section groups together some general advice on getting the best out of CINEMA 4D; it’s like an FAQ (Frequently Asked Questions) list without the questions.

Many of these tips and tricks have been provided by MAXON’s CINEMA 4D support teams and are solutions to common problems faced by first-time users. Others, such as how best to brighten a scene, are general advice from experienced 3D users.

Dip into these as you need them and use them as a supplement to the main tutorials in this manual.
Modeling

Getting a cleaner view.

Do you find that all those polygons and points clutter your view when doing precise modeling? When you are modeling an object, the straight lines from polygon edges can often break up the subtle shading which is required to create smooth, flowing surfaces. There is a solution to this problem. Switch to Edges mode and select all of the edges, then select Selection > Hide Selected. If you now switch back to Points mode, only the points will be visible, giving a point cloud display mixed with a shaded 3D view.

Giving flat polygon objects a thickness.

Sometimes you might want to create a simple, flat object by creating some polygons manually. But how can these polygons be turned into an object with thickness? The Extrude function is designed to build upon an object which already has volume, so it creates thickness but deletes the original polygons. When extruding from a flat surface, this leaves you with a hollow container.

The simple answer here is to copy the flat surface before you perform the Extrude command, then, once you have extruded the object, you can paste the original flat model back into the scene.
To finish off the job you should multi-select both objects then select Function > Connect, followed by Structure > Optimize (the default settings are suitable). These last two operations are essential to maintain a clean mesh that does not break apart if you later decide that you want to distort it in some way.

► **Creating holes in Extrude NURBS.**

If you want to create an Extrude NURBS with a hole, or several holes, the splines you want to extrude need to be combined as a single spline.

If your splines are primitives (parametric, without any editable points), you need to first select all of your splines and select Structure > Make Editable. Next, ensure all splines are still selected and select Function > Connect. Your splines are now combined into a single spline which consists of several segments. The original splines can be deleted. If you now drag the spline into the Extrude NURBS, all inner spline segments will be cut out of the Extrude NURBS geometry.

► **Saving time and effort with instances.**

Instances are references to an original object. By using instances you can not only save a lot of modeling time, but also hard drive space. An instance will look at a single object, then duplicate it wherever you want. Anything you change on the original model will be changed on the instance too, in realtime. And, because only one version of the object needs to be stored, this will also reduce the amount of storage space required for the scene file.

Imagine you are modeling a house with many similar windows and pillars and a street in front of it with several street lamps. What if you have duplicated and placed all these objects and now you need to change them? You would have to delete them all, change the originals and then duplicate and place them all again. But with instances you don’t need to worry about this at all. If you create one original window, pillar and street lamp and the rest are instances, all you need to do is change the originals and all the instances are updated automatically. This saves a huge amount of time.
Texturing

- **Use separate materials for NURBS caps and roundings**

  If you have created a logo using the text spline and an Extrude NURBS, you can assign different materials to the caps and fillets for the Extrude NURBS. You could, for example, make a stone text logo with a golden outline. For this you would assign the stone material to the Extrude NURBS and the gold texture would be restricted to the fillets. All NURBS objects have invisible selection sets for their caps and fillets. Materials can be applied to them by using the texture tag’s Selection box in the Attributes manager. C1 = Cap 1, C2 = Cap 2, R1 = Rounding 1 (fillet 1), R2 = Rounding 2 (fillet 2). Be sure to use capital letters C and R.

This logo uses a separate texture for the fillets.

- **Applying materials to polygon selections.**

  Materials can be restricted to specific polygons of an object. First, select the polygons to which you wish to apply the material. With the polygons selected, select Selection > Set Selection. Give the selection set a unique name in the Attributes manager. When you apply a material to this object, enter the name of the selection.

- **Dealing with the missing textures error.**

  If you have created a scene and decide to save or copy this scene elsewhere, the textures will not automatically be saved with it. If you then try to render the scene, CINEMA 4D might give you a missing textures error message. The best way to avoid this is to use the File > Save Project command when you want to save a scene to a different place on your hard drive. This command collects all the textures that are used in your scene and saves them with your scene in a sub-folder called tex.

  When looking for textures, CINEMA 4D automatically looks in your scene folder, a tex sub-folder in the scene folder, and a tex folder in the CINEMA 4D main folder. If you have a big texture library on your hard drive and don’t want each texture to be copied to your scene folder every time you load one, you can specify the path to your texture library in the Preferences.
Lighting

► **To brighten a scene, use fill lights (not ambient light).**

Don’t use ambient lighting to brighten shadows or dark corners in a scene, use fill lights instead. Ambient light reduces the contrast in a scene. Fill lights reveal hidden details because they not only brighten but also shade objects. Plus they can be restricted to an area by using falloff.

► **To darken areas of a scene, use negative lights.**

A big advantage of using a 3D software is that you can bend the rules of reality as much as you like. CINEMA 4D, for example, lets you use negative lights to subtract light from areas which you want to appear darker.

Create a light and enable the Falloff option on the Details page of the Attributes manager (otherwise the whole scene would be darkened). Also on the Details page, set the Brightness to -100%. This will use the color from the General page and subtract it from the scene.

If we subtract white (RGB all at 100%), the scene becomes black (RGB all at 0%). If we subtract yellow (RG both at some percent), the scene becomes blue (RG both at 0%), and so on. Using this trick, we can create shadow where there is actually no shadow. We could, for example, place a light under a table or in a corner of a room to add extra realism to our render.

► **Projecting a texture from a light.**

We can apply a texture to a light to project it onto the scene. We could, for example, apply an animated wavy noise shader to a light which illuminates the ground of the ocean, simulating the caustics cast by the sun and the waves on the water surface. Additionally, we could also make the noise shader visible with volumetric light. This way we get water rays reaching from the surface to the bottom of the ocean.
To make this work we need to create a material, load a shader (or a texture) into the Transparency channel and apply this material to the light. For omni lights use Spherical projection and for spot lights use Flat projection and adjust the projection using the Texture Axis tool. This isn’t only useful for underwater scenes but also for special effects or for projecting images onto a wall.
Animation

The difference between the Model and Object tools

There are two very similar tools in CINEMA 4D which can both be used to move, scale and rotate objects. One is called the Model tool and the other one is the Object tool. Before you start animating it is important that you understand the difference between the two and when to use which.

Left: the Model tool, Right: the Object tool.

The rule of thumb is this: The Model tool is for modeling, the Object tool is for animation.

You notice the difference only when scaling an object, but if you do use the wrong tool, the result can be puzzling. When scaling an object with the Object tool, you are not only scaling the geometry, you are also scaling its axes. And, especially when scaling an object’s axes non-uniformly (stretched more on one axis than another), the result is that child objects get squashed or stretched when you rotate them and you may begin to wonder what is happening to your objects.

If something strange happens to your objects when you rotate them, then there are most likely scaled axes involved. So when modeling, use the Model tool.

It is different for animation, however. If you want to animate the size of an object over time, you need to use the Object tool. When animating, CINEMA 4D looks only at the object axes, not the geometry. So here it is necessary to scale the axes. If you accidentally use the Model tool and try to record keyframes for an object’s size, nothing will happen because the axes aren’t scaled.

So if you are animating the size of an object and you realize that nothing is actually recorded, you probably have the Model tool selected. For animation, use the Object tool.

Changing the animation length.

If you are creating an animation which you want to be longer than the default length of three seconds, you need to change the document length. Open the Project Settings dialog from the Edit menu and type in a different Maximum value. In the Project Settings you can also change the frame rate for your animation.

Switching between different cameras.

Using camera cuts to switch between different viewing angles (for example, switching from a wide shot of a group of people to a close up of one person) makes an animation more interesting. But you don’t need to render separate files and compose them afterwards in video editing software. All your camera cuts can be planned directly in CINEMA 4D.
The Stage object can be used to do camera cuts in an animation.

Set up all the cameras and create a Stage Object (Objects > Scene > Stage). In the Attributes manager, drag-and-drop the first camera object into the Camera input box and record a keyframe for this parameter (Add Keyframe from the context menu). Then, when you want to switch to another camera, drag-and-drop the second camera object into the Camera input box and record another keyframe.

**Animating a camera along a spline.**

You can make a camera move along a spline very easily. First, create a spline and a camera object. Right-click (command-click on Mac) on the camera in the Object manager and select New Expression > Align To Spline expression from the context menu. This will place the camera at the beginning of the spline. To make the camera rotate according to the spline shape, enable the Tangential option.

At the moment the camera isn’t moving. To make it move you need to animate the Position parameter in the expression tag. On the first frame of your animation, add a keyframe for Position=0% and another keyframe at the end of your animation with Position=100% (Add Keyframe from the Position context menu). This has the advantage that you can directly control the camera speed using F-Curves or additional keyframes.

If you want to control the banking of the camera, necessary perhaps if your camera is doing loops, you can use a second spline as a rail spline. So, in the case of a circular spline, simply duplicate the circle and make it a bit bigger than the first spline. Drop the rail spline into the Rail Path input box of the expression tag. The camera’s X axis is now always pointing at the rail spline. None of this applies only to cameras, any object can be moved along a spline in this way.

Note that in Release 8 the Align To Spline expression replaces the old Align To Spline track in the Timeline of earlier releases of CINEMA 4D. This is also the case for the Vibrate and Align To Path tracks, which are both expressions now.
Rendering

▶ **Be lazy.**

The best tip we could possibly give you is to use the lowest settings possible to get the acceptable level of quality.

If the audience won’t see it, don’t bother with it!

This should apply to all of your 3D work, not just the rendering. A good example would be the chap who sent his scene to our technical support department because, he said, it was taking too long to render. He had spent months carefully modeling and texturing architectural details on a building for a video presentation, but his frames were taking around half-an-hour each to render.

After we examined the scene, it turned out that he had modeled every single detail imaginable, right down to the screw heads on the door hinges. Now, if this were a still image, this level of detail might be needed, but at video resolution his immaculately modeled screw head equated to a tenth of a pixel wide. In fact, his beautifully carved door frames were only two pixels wide! The bottom line is, it would have looked identical whether he spent two weeks modeling the doorway with detail or whether he just roughly positioned a few cubes to represent them.

A few minutes of thinking and planning before you get down to the work could potentially save days, weeks, months of unnecessary work further down the line.

Work smarter, not longer!

▶ **Selecting the correct antialiasing.**

Antialiasing is another good example of where time can be unnecessarily wasted. CINEMA 4D offers three levels of antialiasing: None, Geometry and Best.

Setting antialiasing to None will leave you with the original jagged edges on any diagonal lines, while the Best setting will give you a perfect quality image every time.

Best does have one disadvantage, it requires a lot more render time to smooth all of the edges. Geometry antialiasing should be used wherever possible, which is why it is selected by default. Geometry antialiasing tries to be as clever as possible by smoothing only those edges which really need it.

While Best antialiasing will use the sheer brute force of antialiasing every pixel in the image, Geometry antialiasing is able to detect which edges will give jagged lines and how jagged they will be. This way it can leave any vertical and horizontal lines as they are, ensuring they remain crisp and untouched, while applying antialiasing selectively to all other lines.
Another reason to use the Geometry setting is that it requires very little extra render time compared to no antialiasing at all. Typically, you’ll find that less than a quarter of the scene needs any kind of antialiasing.

There’s no such thing as a free lunch, though. The limitation of Geometry antialiasing is that it can work only on objects directly seen by the camera. This means any objects reflected in another object, or any behind a transparent object, will not be antialiased. For scenes where large reflections play a role (a hall of mirrors, perhaps, or a bathroom scene) you will need to switch to Best antialiasing to remove the jagged edges.

So, in short, use the None setting for test renders, Geometry for everything you possibly can and Best for scenes with prominent reflections and transparencies.

Lastly, it is worth noting that any scene in which you are using motion blur can usually have antialiasing disabled completely. Motion blur in CINEMA 4D will very slightly move the camera with each pass, thus eliminating any jagged edges from diagonal lines.

**Field rendering.**

Virtually every TV set in the world offers a feature known as fields. Most people know that a TV has a frame rate of 25 or 30 frames per second (fps) depending on whether you are working towards the PAL or NTSC standard. The frame rate isn’t the end of the story though. Each frame is split into two interlace fields, giving a total of either 50 or 60 fields per second. These fields are interwoven to produce a complete picture.

For the first half of a frame, the upper (or odd) field lines will be displayed, then, on the second half of the frame, the lower (or even) fields will be displayed.
Because each image lingers for a moment before fading from the screen, we see both fields at about the same time.

Normally, the flickering between the fields can barely be seen unless there is a crisp, stationary horizontal line in the image, such as you might get in the composites of broadcasts and sporting events.

In CINEMA 4D, by enabling field rendering in the render settings you are able to take advantage of these fields to double the smoothness of the animation being displayed while encountering virtually no increase in render times.

Note that field rendering is not always a good thing. You should not use it when you are rendering for display on a computer screen or when trying to achieve a stylized look, such as grainy 35 mm film or hand drawn animation.

▶ **Do frequent test renders.**

Test renders should be an important part of your everyday 3D work. Until computer processors are fast enough, the real-time display in the viewport will never match the quality or accuracy of a render. This means that occasionally you will have to do a few test runs to check that everything appears exactly as you want it to.

Take radiosity, for example. Radiosity renders can take anywhere from a few seconds to a few hours. To check that you have the right lighting levels you will want a very quick render. In such cases it doesn’t matter if lines have jagged edges or if it is not at the full resolution you intend the audience to view it at. These finishing touches are only there to please the audiences, you will be able to tell if a scene is correctly lit regardless of how rough it looks.

We recommend that you make use of the rendering presets and adapt them to your own work. You will find that the quick render preset has all of the time-consuming options such as antialiasing, radiosity and caustics disabled.
Make use of the Render Region button in the top toolbar. The Render Region command can be used to render a small section of the view to full quality. This might be especially useful if you have an area at the bottom of your scene which you wish to view, but don’t want to have to wait for the rest of the image to be processed first.

**Selecting the right file format.**

Creating scenes and rendering images isn’t the end of the production process in most cases, there’s still the question of how do you physically show it to your audience. Should you use a common codec to enable all computer platforms to view an animation, or should you chose the superior qualities of MPEG-4?

At the render stage the answer is neither. Avoid rendering your images to any video file format, especially compressed ones. Although CINEMA 4D provides you with a wide range of video file formats to render to, it is wise to always render your video as a series of still images.

There are several reasons for doing this. Production reliability, for a start. If you are rendering to a video file, most compression codecs will not allow you to play back the file until it has finished being written to. There are several technical reasons for this which we don’t need to go into, but the bottom line is that you can’t go back and check any rendered frames until the very last one is done and the video file has been saved.

Also, imagine what would happen if you left a computer rendering a large animation for a week then found out that at 99% completion the hard drive was full or that there had been a power cut. In such cases the entire video file would be garbage and very difficult to salvage, if at all possible.

Then there’s giving yourself options. If you render as a series of still images you will have the choice later of what format you want to save the video. By keeping the raw footage you can spend time experimenting with different levels of compressions to get a good balance of quality and file size. If you had already saved this as a compressed file then you would either need to re-render it to change the compression, or you would have to re-compress it, lowering the quality even further.

And we shouldn’t forget convenience. If you render each frame as an individual file then you have the ability to stop the render at any point and resume it later when it is convenient for you. After you have stopped the rendering process you may resume from where you left off by setting the render range to begin with the last full frame that rendered. You should keep in mind that whichever frame was currently being rendered will be lost when aborted and must be rendered again from the start.

If you are rendering as a series of still images you will of course require some other software which is capable of stitching these frames together into a sequence. Virtually all video editing software is able to import a sequence of still images; your video editing package may refer to it as a DIB sequence.
Use the NET Render module to reduce production times.

With the NET Render module you will have the very powerful option of using an entire network to render your images. This could be just a few spare machines you have laying around your office, or an entire university campus on the other side of the world with 1,000 computers. NET Render will enable you to harness the power of more computers than just the one you’re working on.

Even if you have just a couple of old, slow workhorses sat in the corner, you should consider adding them to your ‘render farm’. If they render only a handful of frames, that’s still a handful less for your main machine to render.

Look before you leap.

Before you set your scene to be rendered, whether on a single machine or via NET, always triple check that you have the right settings and all required plugins and textures are available. There’s nothing more annoying than setting a feature film off to render while you go on holiday, only to return to find you forgot to turn off the gigantic lens flares you were messing about with late one night.

Hardware

Cutting edge software works best in conjunction with cutting edge hardware, but keeping this up to date can be quite expensive. Whether you’re deciding which computer to buy, upgrade or use, the following points might help you decide.

When it comes to choosing a CPU it’s a simple case of the bigger the better. The processor will affect both the viewport speed and the time it takes to render a final image. 3D art is the most processor intensive form of art there is, so select a processor with as many GHz as you can afford.

You should also fit as much physical RAM as you can afford to. The worst offenders for memory usage are architectural scenes and detailed product designs. These types of scenes often need to have every single detail modeled accurately. If you find yourself running out of memory while rendering, then in most cases your system will begin using the hard drive as additional ‘virtual’ memory. This should be avoided if possible because your hard drive is thousands of times slower than real memory.

One of the most commonly asked questions we get is whether or not a graphics card will make a difference to the rendering speed. The answer is no. A graphics card will make a difference only to the 3D viewport speed. If you have an old graphics card, or a cheaper low-end one, then you should consider using CINEMA 4D’s built in software display. This will use your regular processor to calculate the 3D display in the viewport rather than any special 3D chips on your graphics card.
If you are lucky enough to own a modern graphics card then you should consider enabling the OpenGL hardware display option in CINEMA 4D’s Preferences. By doing this you will shift most display calculations from the main processor onto the graphics card’s processor. As these have been designed specifically for showing 3D content, they will often be able to give very high frame rates and crisper images than the software calculations. CINEMA 4D will use any OpenGL compatible graphics card, whether it be a consumer level card or a high-end professional workstation card costing thousands.

Operating systems is something else we are often asked about. Should you upgrade to the latest version? A simple rule of thumb is to use the most recent operating system you have available for your hardware. New operating systems will offer greater support for hardware and software features, enabling more speed to be gained than older system software.

Which platform should you should buy, PC or Mac? This is a question only you can answer. CINEMA 4D will operate identically on all supported platforms.