NEW

FREE 23GB OF DIGITAL ASSETS

The 3D Art & Design ANNUAL

Take your 3D projects to the next level & transform the way you work

Welcome to

The 3D Art & Design ANNUAL

The world of 3D art and design grows exponentially every year and has been doing so for a long time now. Whether it's in the modelling, texturing and real-time rendering you see in videogames, the blockbuster visual effects you see in movies, the clever invisible effects that are the building blocks of many TV shows or anything in between, the impact of developments in the visual effects industry can be easily identified throughout the entertainment, product design and medical fields. Whether you're a professional working in the industry, a student or enthusiast, or just interested in the limitless possibilities that computer graphics present, The 3D Art & Design Annual – brought to you by the team behind 3D Artist magazine – is packed full of expert tutorials for you to absorb, written by some of the greatest artists in the field and covering every aspect of the 3D production pipeline from concept to the final render. On top of this, we've provided you with an astonishing assortment of free 3D resources for you to download, including hours of video tuition, 3D models to use in your work and much more.

The 3D Art & Design ANNUAL

Imagine Publishing Ltd
Richmond House
33 Richmond Hill
Bournemouth
Dorset BH2 6EZ
☎ +44 (0) 1202 586200
Website: www.imagine-publishing.co.uk
Twitter: @Books_Imagine
Facebook: www.facebook.com/ImagineBookazines

Publishing Director
Aaron Asadi

Head of Design
Ross Andrews

Editor
Steve Holmes

Senior Art Editor
Greg Whitaker

Assistant Designer
Harriet Knight

Cover image
Fausto De Martini

Printed by
William Gibbons, 26 Planetary Road, Willenhall, West Midlands, WV13 3XT

Distributed in the UK, Eire & the Rest of the World by:
Marketforce, 5 Churchill Place, Canary Wharf, London, E14 5HU
0203 787 9060, www.marketforce.co.uk

Distributed in Australia by:
Network Services (a division of Bauer Media Group), Level 21 Civic Tower, 66-68 Goulburn Street, Sydney, New South Wales 2000, Australia Tel +61 2 8667 5288

ISBN 9781785461675

Part of the 3DArtist bookazine series

IMAGINE PUBLISHING

Contents

08
Behind the software

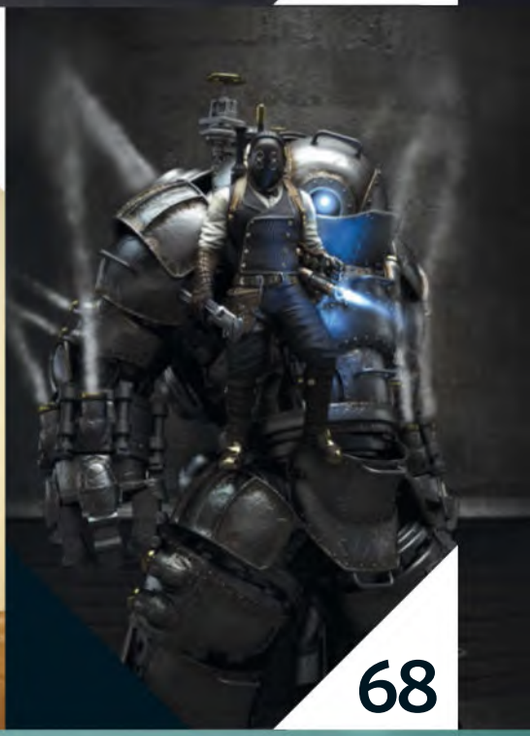
Modelling

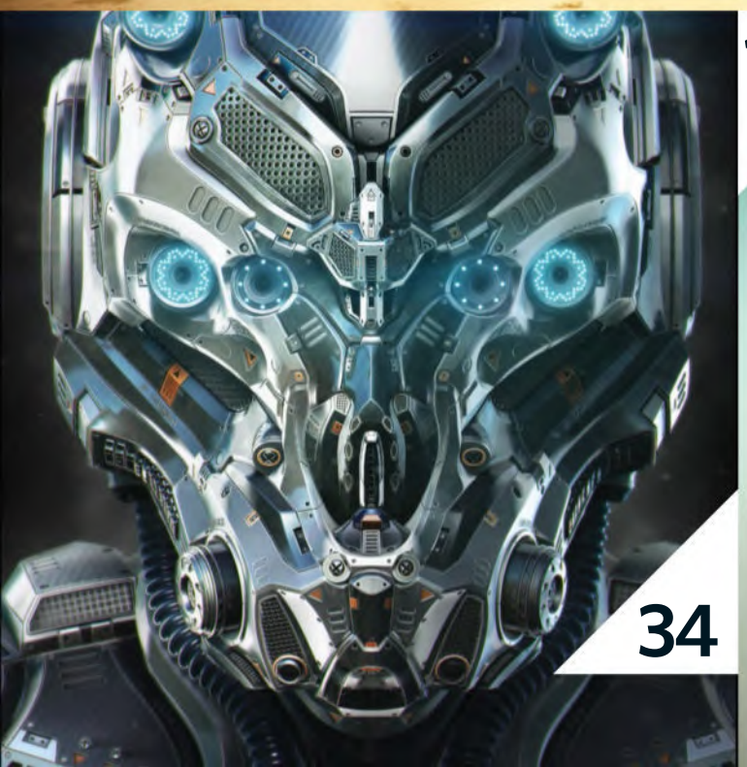
Texturing

Visual Effects

Rigging & Animation

Lighting & Rendering





INSIDE THE SOFTWARE

We interview the developers behind the leading CG software used to create everything from Oscar-winning feature films to stunning games graphics

Going from computers that took up entire rooms and were capable of only the crudest of CG images, to the personal workstations of today that run software so powerful that anyone can create animations in their own home, has been no easy feat. It has taken many years and the dedicated work of thousands of artists who are, even today, pushing the limits of what computer graphics are capable of.

However, while we may see a list of modellers' and animators' names scroll past during a film's credits , the software developers themselves are often notably absent. Yet, it's thanks to their pioneering work that any of this has been achieved at all. Though we rarely think about it, we likely see something created using the software they've developed every single day of our lives.

To find out more about the people behind the tools, we spoke directly to the developers of the world's leading CG programs, learning how they go about creating and building this new, incredible technology that stands as the artist's latest, most highly sophisticated paintbrush.

Interviewees

Autodesk
Trevor Adams
Gordon Bradley
Maurice Patel

Blender Foundation
Campbell Barton

Side Effects Software
Ed Lam

The Foundry
Matt Plec
Simon Robinson

Time Machine was made using Maya, mental ray and Photoshop by Aleksandr Kuskov, **www.behance.net/AleksCG**

> The Holy Grail for a designer is developing something that is easy to learn, but challenging to master

Trevor Adams, Senior user experience designer at Autodesk

Autodesk

Bio Founded in 1982 by John Walker and Dan Drake, Autodesk is a world leader in 3D design, engineering and entertainment. Autodesk delivers products that aid in the work of over 10 million customers, and has been included in *Fortune*'s list of 100 Best Companies to Work For. Along with the popular 3D software Maya, Autodesk also produces tools such as its flagship computer-aided design software, AutoCAD
Software Maya, 3ds Max, Mudbox, Softimage
Known for Having developed software that has been used in many Academy Award winners for Best Visual Effects. Recognised by the Academy of Motion Picture Arts and Sciences with two Scientific and Technical Achievement Awards for Mudbox and FiLMBOX, the foundation of Autodesk MotionBuilder. Autodesk scientists and technologies have been similarly recognised in past Scientific and Technical awards ceremonies
Interviewees **Trevor Adams** Senior user experience designer. Has been working at Autodesk for eight years in several roles, and started as a quality assurance specialist testing the software
Gordon Bradley Senior software architect. Has been working at Autodesk for over a decade
Maurice Patel Entertainment industry manager

How do you decide what features to put in your latest software releases?

Gordon Bradley I think that we definitely have creative freedom to explore future ideas ourselves as software developers, but a lot of the ideas and suggestions for new features really do come straight from the artists.

I think that when you look at the strength of 2015, it really has come out of asking people how we can improve the software. From there, we explore, we prototype, we play with a few options and we get feedback on them to work out the best way to tackle implementing each new feature. It really is a great process and I think that's something Autodesk has been getting right for a long time – working out how to involve the artist and the community all the way through development. That's kind of what leads us, I believe, to some of the really strong line-ups that we have today.

Can you tell us a little about tools such as Mudbox and XGen? What do you need to do when you take on technology like this to get it into the hands of artists?

Maurice Patel The thing is, with technologies such as Mudbox, which actually came out of production technology in Weta, and XGen, which was production technology in Disney, is that very often our big customers will develop a toolset which is highly specialised, but it's not at all user-friendly. They're often very interested in getting this technology into the hands of other artists, so we partner with them to help them.

I remember when we announced that we'd signed a deal with Disney for XGen and everyone was asking where it was and why it wasn't in the next release. It takes years of work to make these tools user-friendly – there's actually a lot of expertise involved in taking these technologies and making them better suited for a broader audience rather than just specialists within huge companies.

What are your main goals when working on software such as Maya?

Trevor Adams The Holy Grail for a designer is developing something that is easy to learn, but challenging to master. It's like picking up a musical instrument like a guitar: you can quickly just strum away at some cords and make some noise.
Gordon Bradley This is something we've said in Maya for a long time – our job is to hide the maths. That's what it comes down to. Over Maya's history that's what we've constantly been trying to do. Hide the maths and let the artist focus on what they want to focus on, which is really telling a story, whether it's film, television or game.

What have been the main challenges and successes of your job developing Autodesk's solutions over the years?

Maurice Patel Honestly, you could write a book about it! Ultimately one of the things that we're very proud of is that we've had a lot of successes. I mean, we've won seven Academy Awards for the technology, whether it's Maya, Flame, or Mudbox and MotionBuilder.

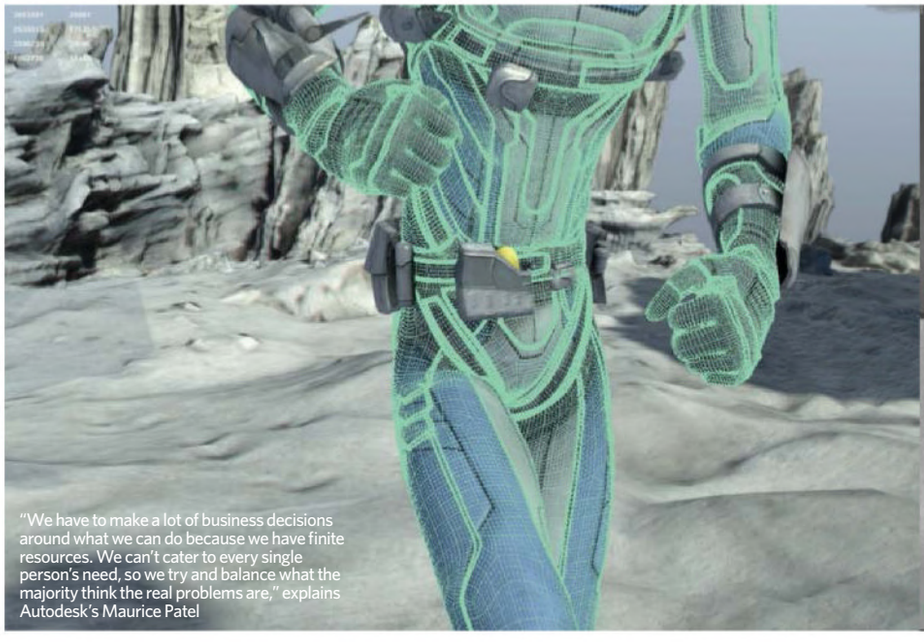
The challenges are many. When you're a technology developer and people want new things, we have to really balance supporting what we would call legacy versus developing something new.

We did this in Flame – it had a very distinctive user interface. All the buttons were placed in very specific areas and were designed to be driven with a

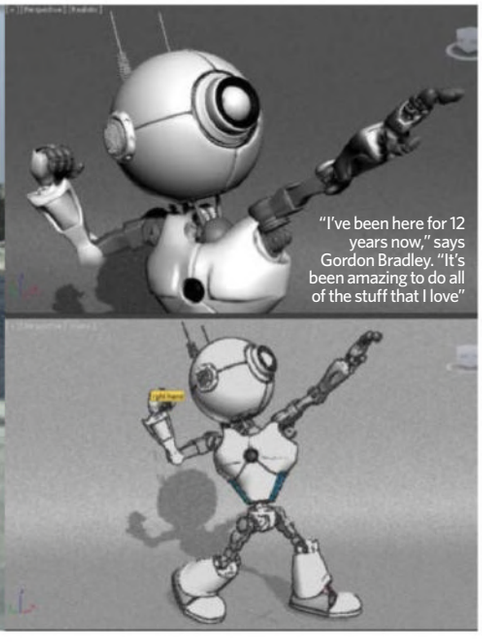
THE PERKS OF THE JOB

"I think the biggest reward for me is hearing from artists," says Gordon Bradley, senior software architect at Autodesk. "When they've picked up a tool that you were a part of and they talk about what they did with it and how much better it made their working life, that's why I'm here. It never fails to make me smile, especially if they don't know that you worked on it."

"Definitely," agrees Trevor Adams, senior user experience designer. "You do feel like you're part of it, even though you're essentially behind the scenes. Even reading articles like in **3D Artist** about the new features that are coming out is rewarding in itself, to realise that it's real, it's in the application and it's no longer being tested. To have so much positive feedback inspires you to work even harder because people are paying attention and appreciating it."



"We have to make a lot of business decisions around what we can do because we have finite resources. We can't cater to every single person's need, so we try and balance what the majority think the real problems are," explains Autodesk's Maurice Patel



"I've been here for 12 years now," says Gordon Bradley. "It's been amazing to do all of the stuff that I love"

WACOM tablet using muscle memory, just like touchtyping with a keyboard. You could run this product without ever looking at the user interface. This is another product that has been around for 20 years and for its twentieth anniversary we redesigned it to make it more modern, but there was huge resistance at the beginning, of course. These are people who can sometimes charge between 500 to 1,000 dollars an hour for their work and all of a sudden you're impacting their productivity by changing what they're used to!

How do you decide what features to let go of if you have developed something that does manage to do things better?

Gordon Bradley This is something that historically we didn't do very well and it resulted in a lot of complexity in the product. Our goal is never to pull the rug out from right under people's feet, but I think that by and large there are so many advantages to moving to the new stuff. As soon as the artists let us know that they're ready to move, we do try to keep it as streamlined as possible.

Trevor Adams Absolutely, that's where the beta program comes into effect. For example, we reintroduced Multicut, a way to cut up your polys in Maya, but I could probably count about five or six different tools in Maya 2014 that essentially do similar things. Our goal with the 2015 release was to change this. We focused on this one tool for several months to make sure that the new version of Multicut comprised of all the features of the previous tools and then finally allowed ourselves to start deleting things. In 2015, you don't have Interactive Split or a Split Polygon tool anymore.

We have to ask – why did Autodesk decide to discontinue Softimage?

Maurice Patel Typically, these types of decisions are business decisions and are not made by the developers, although their analysis on the feasibility of various plans is a very critical factor in determining the right course of action to take. This decision was not really made from a perspective of merging technologies, but from a need to free up more resources to increase the rate of development and innovation on our more popular products, such as 3ds Max and Maya. It is our intention to take elements from Softimage and incorporate them into 3ds Max and Maya where appropriate.

What are you most looking forward to from future releases?

Trevor Adams What I'm looking forward to in [future releases] is seeing people start realising that modelling in Maya is just more fun, and what I look forward to in the future, being the modelling designer, is to continue on that front. I want to make Maya that application you can just open up – you don't really have to read anything about the application, you just pick it up and start playing around and you start building models. I'd like people to be able to get that kind of instant gratification from using our software.

DEVELOPING V-RAY

Lead developer and Chaos Group co-founder Vlado Koylazov discusses his experiences creating and developing the major renderer, V-Ray, which started life as a small raycasting engine Koylazov wrote to obtain volumetric shadows.

What are the key things you need to keep in mind when building a good renderer?

It depends on what the renderer is used for. For example, a game engine could be a good renderer for its purpose, which is real-time interactive graphics. REYES-style renderers were favoured for a long time for visual effects because of their ability to efficiently render very detailed geometry and camera effects like depth of field and motion blur. Raytracers are good for photorealistic rendering and so on. In my opinion, though, a good renderer should be functional in a wide variety of situations.

The main challenge is that there are no ready solutions for many of the things that we want to implement, so we need to do quite a bit of research to find out good ways of solving particular problems. Specific things that I am particularly proud of are the light cache and V-Ray's proxy geometry system. Both have proven to be very successful approaches for GI and heavy geometry rendering to the extent that they are now considered a must-have for any render engine.

Is there any challenges specific to developing renderers that may not be encountered when working on other computer graphics software?

There are difficult problems to solve in any field, but the main challenge with rendering is understanding the underlying theoretical framework and how the numbers come together to form a final image. Getting the most out of the available hardware, CPUs or GPUs has also proven to be a major challenge.

V-Ray 3 for Maya Beta has just been released © Nick Gaul

Autodesk's switch to providing software rental options "allows our developers to be even more responsive rather than focusing on big, annual releases," says Patel. "It allows us to be more flexible creatively."

Blender / Blender Foundation

Bio A Dutch public benefit corporation, created to support and facilitate projects on **www.blender.org**
Software Blender
Known for Blender is regarded as the most sophisticated open-source 3D software in the world. It is actively developed by hundreds of international volunteers, including artists, VFX experts, hobbyists, scientists and more
Interviewee Campbell Barton Active with Blender since 2003 and lead developer since 2007

What exactly would you describe as a typical day at work?

Mostly I start by checking our bug-tracker for any new issues and check on any changes made to Blender overnight. From there I'll move on to development – I do a lot of bug fixing and reviewing the code, but also general improvements or features.

There are so many ideas flying around that most Blender developers get bombarded with requests from users. Often, by the time we get around to working on a new feature, it's been on our to-do list for a while. We get a lot of freedom to develop areas we find interesting and that's one of the reasons I really love to work on Blender – but that's balanced with the need to keep the software working and stable, which takes up our time too.

A huge part of what makes Blender special is that it is free and open-source. How does this impact on you as a developer? Also, might we ever see a future where this isn't the case for Blender?

This is hard for me to answer, as I have never done any serious closed-source development. I'll just say that I really like the openness and transparency that we have with other developers and our user-base. We can ask users who want to test a new feature to test a nightly build without them needing to be a part of some kind of beta-program, while users who report bugs even send in fixes occasionally because they are able to download the source code and build upon it themselves.

As for Blender one day getting locked up – in practice, it would mean getting the hundreds of people who own copyright on our code to agree. There's a lot written on this and you could ask the same question for Linux, Firefox or Wikipedia – but users, rest assured, we'll stay free and open! This is fundamental to the software.

> "...you could ask the same question for Linux, Firefox or Wikipedia – but users rest assured, we'll stay free and open!"
>
> **Campbell Barton, Lead developer at The Blender Foundation**

We loved the new website and plans for 2014/15, as well as Project Gooseberry. Can you tell us more about the changes you're working on implementing today?

For the UI we have long term plans to investigate tool access and the toolbar. Some features in Blender aren't very obvious, so we need to improve on that, as well as lots of smaller changes we do as a part of regular development with guidance from our UI team. Today I'm looking to develop some mesh modelling tools to intersect selected geometry – a little like booleans that can be used in Edit mode without setting up modifiers.

What are the best ways that 3D artists and users can help with the future of Blender?

I could recommend donating to the Blender Foundation or buying a t-shirt, but while this does help very much, I think its a short-sighted answer.

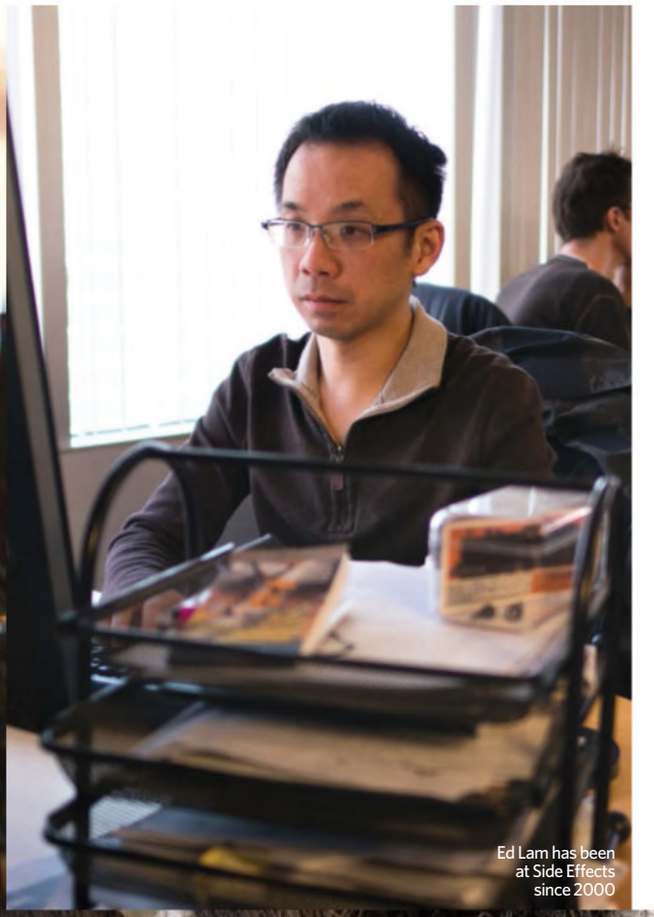
I'd suggest to start small and get involved with the online community, even asking good questions on sites like **http://blender.stackexchange.com** helps. Answering questions, making tutorials and improving wiki documentation helps too, but not everyone can do that to begin with.

Another way to help is by funding development. If you miss some feature, it may be possible to have it written as an add-on or extension. My personal aims for the future are to help with getting more developers involved – this is something that we desperately need if we are to continue growing.

What are the top features you think the best CG software should always have?

Smooth workflow When you perform actions frequently, having to be prompted with dialog boxes each time can really slow you down.
Interactive feedback and previews When artists are changing values for materials, simulations and so on, they need to get quick feedback so they aren't just guessing and having to wait a long time to see results. For areas such as physics simulations this isn't easy, but it's still something to aim for.
Consistency With 3D applications in particular there are many similar concepts mixed together. Consider selecting a vertex. That doesn't just apply to meshes, but also the Animation Curve editor, UVs, image masks and key frames in the dopesheet. If the user has to learn different rules for each one, it takes longer to get used to and muscle memory from one area won't apply to another.
Extensible While the base set of tools should cover common use cases, in my experience there are many special cases that come up when working on 3D projects, so it's important that applications can be extended. While all of the popular 3D applications enable this at some level, they vary considerably in their approach and the kinds of extensions they support.

On the software side, I'd really like to improve on Blender's extensibility. This means better integration with studio pipelines as well as the ability to use Blender as the basis for other tools.



Ed Lam has been at Side Effects since 2000

WHAT IT TAKES TO BE A SOFTWARE DEVELOPER

From Blender's Campbell Barton

"The obvious one for any developer is to be a good problem solver. Beyond that, here are some qualities I think are more specific to computer graphics:

Design Even though many developers wouldn't consider themselves UI/UX designers, often we need to expose complex functionality to users without just presenting a grid of options. Getting the balance right between user control and ease of use isn't easy.

Maths There's no denying that computer graphics can involve fairly advanced maths, so it is an advantage to have a good background here. However, in the same breath I think some people overestimate this. There are many areas of computer graphics software that don't require really advanced maths and developers can learn some along the way when needed.

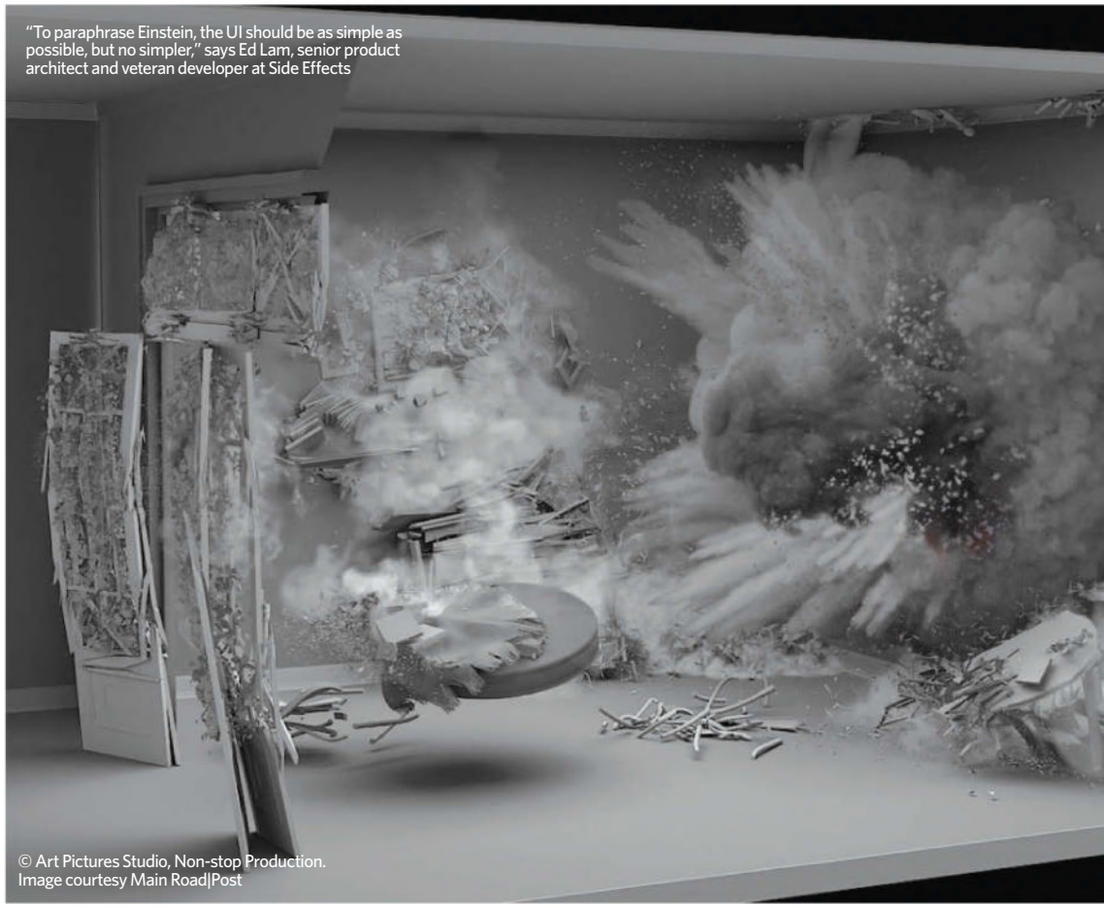
Communication Considering a lot of open-source software is written by online teams, developers have to be able to communicate well so that the project runs smoothly. It's important that the developers take the time to understand the needs of users and can engage with them to help get the best results.

Discipline It seems many areas of software development are moving away from low-level languages like C/C++ to Java/.NET and scripting languages, however with computer graphics, the performance benefits of lower-level languages often can't be ignored.

Images from Project Gooseberry, a project in which the Blender Institute will bring together 12 indie studios from around the world to contribute to an animated film

Houdini / Side Effects Software

Bio Starting out at a pioneering company in the emerging world of computer graphics, Omnibus, Side Effects was born from Kim Davidson and Greg Hermanovic's desire to bring 3D graphics to a wider audience with PRISMS, a procedural graphics application, which laid the groundwork for Houdini
Software Houdini
Known for Twice recognised by the academy of Motion Pictures, Arts and Sciences for Houdini and it's procedural based technology. Houdini has been used in some of the most VFX-intensive films of recent years, such as *Ender's Game, Gravity, Pacific Rim* and *Star Trek Into Darkness*
Interviewee Ed Lam Senior product architect and veteran developer at Side Effects



"To paraphrase Einstein, the UI should be as simple as possible, but no simpler," says Ed Lam, senior product architect and veteran developer at Side Effects

What inspired you to start developing computer graphics software at Side Effects?

I was drawn to computer graphics ever since I first started learning programming as a kid. I have fond memories of writing programs on the TI-99/4A that would draw colourful patterns on the monitor, as well as writing a simple drawing program with my friend on his VIC-20. I enjoyed the feeling of creating something visual with the computer, and being able make it interactive was icing on the cake.

During my university internships, I tried a few different areas but I came back to my first love in the end, computer graphics. I had briefly worked at another, much larger 2D graphics company during an internship, so when I graduated I was looking for someplace smaller, where I could have a greater impact. Side Effects Software fit the bill perfectly. The team at the company was small and they developed an incredible program called Houdini that was used to create visual effects on blockbuster films. It was very cool for someone embarking on their first real, full-time job in 2000. Here I am today, still at Side Effects.

What is it like as a developer at Side Effects, and what are your goals for the software?

The culture at Side Effects is extremely creative. The main sources of inspiration come from artists and technical directors, as well as each other on the team. Side Effects has a great rapport with the users. I rarely go a day without talking to an artist or TD. The developers are constantly working together on different things and share the challenges they face in their respective areas of the package. This creative process mingles a lot of ideas together and provides lots of food for thought. Beyond the usual public feedback mechanisms and support channels, we also hold special discussion events that we call Houdini Forums. They are held typically twice a year in London and Los Angeles. A cross-section of companies are invited with a good mix of competencies to solicit how we're currently doing and to act as sounding boards for wild ideas.

We're not shy about replacing old parts of the architecture in major projects that sometimes span multiple years and releases. For example, Houdini 12 sported a rebuilt geometry core and Houdini 13 introduced a new particles architecture. This is what keeps the software fresh and scalable after all these years. Since Side Effects is a private company, we can afford this important luxury due to the fact that we are fully in charge of our destiny. Ultimately, our goal is to do right by our user base.

> "Don't just learn in a vacuum. Learning from tutorials gives the basics, but it isn't a substitute for trying things out yourself – don't be afraid to ask the community for help in our web forums"
>
> **Ed Lam,** Senior product architect and veteran developer at Side Effects

For you, what are the easiest ways in which a beginner can start to learn the software? What methods would you suggest?

For starters, go to **www.sidefx.com**, click on the menu at the top and go to Learning>Getting Started, which has a good set of introductory lessons. If you scroll down to the bottom of this page you will find a link to PQ Houdini. Artist extraordinaire Peter Quint has over 100 tutorials on his Vimeo channel.

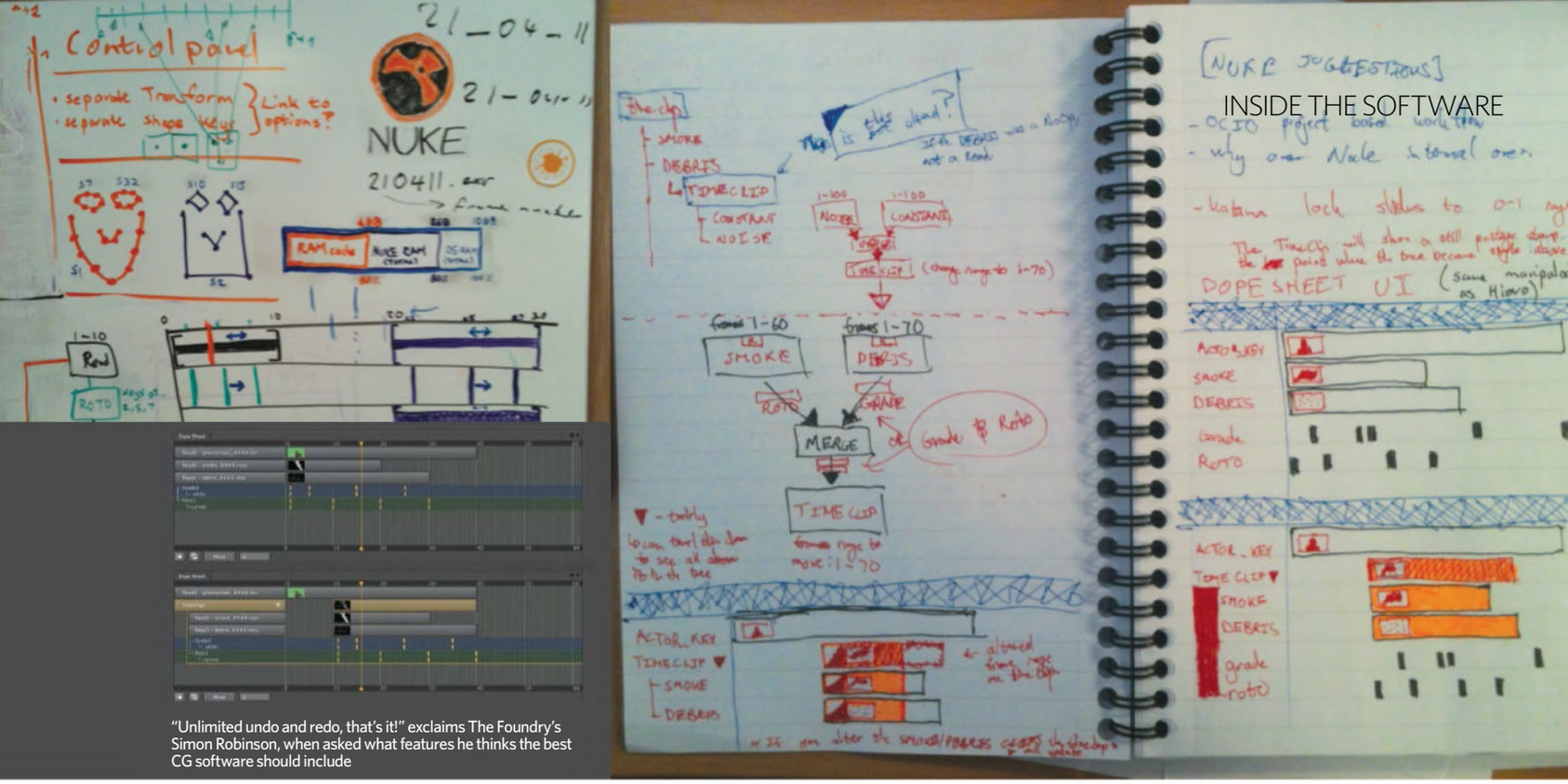
There are more tutorials under Learning>Tutorials. In addition to the usual ones, the Master Classes are unique. They are advanced, in-depth topic videos presented by the software developers themselves.

Don't just learn in a vacuum. Learning from tutorials gives the basics, but it isn't a substitute for trying things out yourself – don't be afraid to ask the community for help in our web forums.

How do you go about deciding what new features to add to Houdini?

The power of Houdini lies in its flexible procedural nature. Whatever it is that I'm implementing in Houdini, I'm always thinking of how to not only make that feature useful for the task at hand, but also to ensure that it is extensible so artists can take it to the next level in ways that we can't envision. We're often taken by surprise at how imaginative the Houdini community is, using tools in ways that we'd never even thought of in the first place.

I believe that software needs to be heavily used before all features can be discovered. We're not perfect, so being open with the community and making daily builds available really helps to improve the software. As a programmer, it is vital to have strong software engineering skills. With a large code base like Houdini's, it's always challenging to ensure that complexity is kept at bay while adding features.



"Unlimited undo and redo, that's it!" exclaims The Foundry's Simon Robinson, when asked what features he thinks the best CG software should include

NUKE / The Foundry

Bio A global developer of computer graphics, visual effects and 3D design software for the design, visualisation and entertainment industries with offices in London, Los Angeles and Silicon Valley
Software NUKE, MARI, MODO, KATANA, FLIX
Known for Software used for Oscar-winning feature films, television, games and design. Awards include wining the 2013 HPA Engineering Excellence Award for FLIX and an Academy Technical Achievement Award for its work on KATANA
Interviewees Matt Plec Senior product designer, originally part of the commercial development team on NUKE
Simon Robinson Chief scientist, from a background in computer algebra systems

Do you feel it's possible to develop computer graphics software without being a 3D artist? How is this done?

Simon Robinson The engineering of a complicated piece of software needs lots of different disciplines, ranging from the extremely technical details about software architecture, to the look and feel and user experience. You can get a lot done without being an artist yourself, but you certainly can't do it without knowing how users need to work, and you can't do it without great respect for what they produce.

Matt Plec When you spend time working in a piece of software – and some of our customers spend a lot of time in our software – you notice small, but really important things. Our engineers need that kind of time to do the development work, too, so the feedback and interaction with customers, product designers and creative specialists is invaluable.

What's changed about 3D software development since you began?

Simon Robinson These days I do very little coding. I've been replaced by more talented people! At the start of my career, working in production, I was thrown a fresh challenge pretty frequently. I worked long and extremely antisocial hours, always working out how to persuade software to generate an image that was good enough from the required angles for long enough to be convincing. Over the years, as The Foundry has grown up, development has become increasingly organised, so that engineers' time here is more reliably scheduled. There's still a lot of stress and hard work involved though, and the antisocial hours still remain a part of the industry.

Are there any other difficulties you could expect as a software developer?

Simon Robinson We sort and discard our ideas by asking clients what they think and by constantly quizzing them about what it is we're making, who it is meant for, in what market and – because this is a business, at the end of the day – how much the people in those markets think our work is worth. Then within the development process itself, there's lots of debate about the cleanest way of doing something, questioning whether something from another team or another project could be reused. There's a lot of process involved, but there is a lot of experience needed to make the right choices.

We also have some highly technical software. When you open it, it feels a bit like stepping into the cockpit of a 747. You wonder if it really needs all those buttons. It may be intimidating, but it has a purpose. Yet, at the same time there is a greater drive to software being more accessible – can an artist pick the software up within five minutes of opening it for the first time and immediately be productive? The line between simple and complex is a trial and error process and it's constantly evolving.

Do you feel this is a job that could not be possible without feedback or beta testing?

Simon Robinson Feedback is as important as typing the code in the first place. Too many projects become unstuck because developers are overly confident about the brilliance of their own ideas!

How can 3D artists and users help influence the development of NUKE, KATANA, MARI, MODO and your other software?

Simon Robinson Feedback, engage, criticise! Come in for a cup of tea and tell us what you think! Feedback is paramount to our improvements.

Matt Plec In your feedback and criticism, try to help us understand the root of the issue. Sometimes someone says, 'you need to add a button for X', but when we dig deeper and ask why, we find we can actually do something that will solve that problem and much more. The 'why' is really important, all the more so because we're not the ones using the software all day every day.

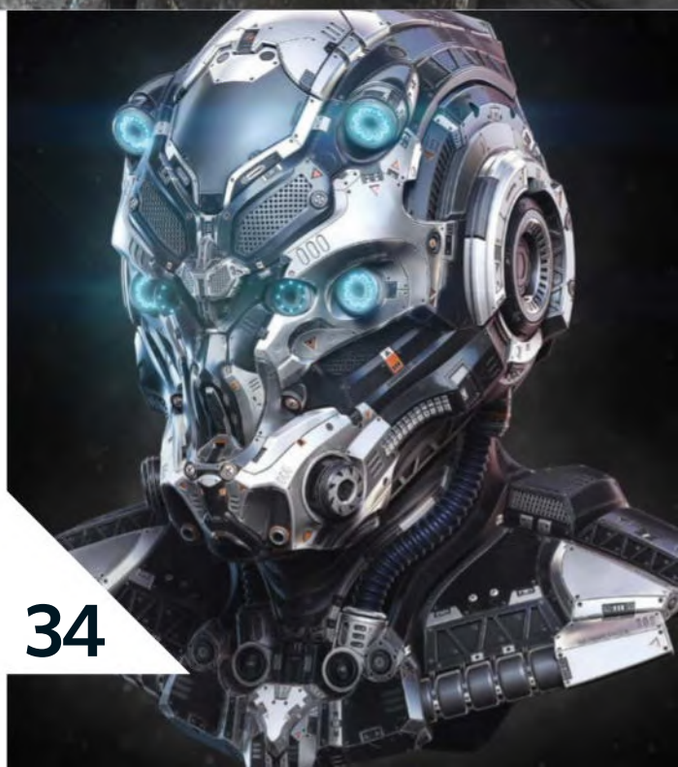
What would you most want to implement at The Foundry within the next five years?

Simon Robinson Quite seriously, if we said confidently that we knew exactly where we would be in the next five years, we would be lying and you should ignore us.

Matt Plec I'll take the bait – I don't want to wait for updates when I'm working interactively but if I have to, I would like to be able to control the render quality more finely than Proxy On/Off. If I can get hold of more machines to do some of the computing, I would like to be able to use them effortlessly whenever or wherever they're available. Iterations should be faster so we can experiment more. If I'm honest, that's just the start of a very long list.

> "You can get a lot done without being an artist yourself, but you can't do it without great respect for what they produce"
> **Simon Robinson,** Chief scientist, The Foundry

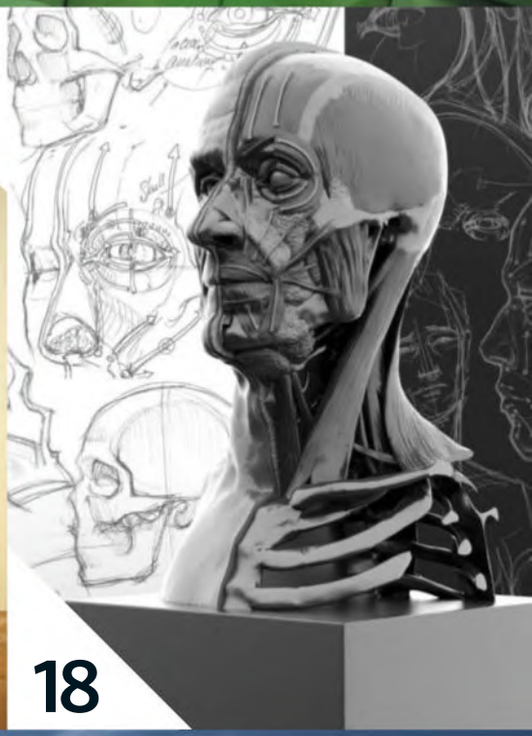
Modelling



56

52

32



18

42

SCOTT EATON
Facial Anatomy Ecorche, 2015

Software
ZBrush

Learn how to
- Locate the important landmarks of the face
- Locate and study the function of the skeletal muscle
- Review the influence of subcutaneous and deep fat deposits of the face
- Construct the eyelids and surrounding orbit
- Shape facial muscles according to their influence on the form and expression of the face

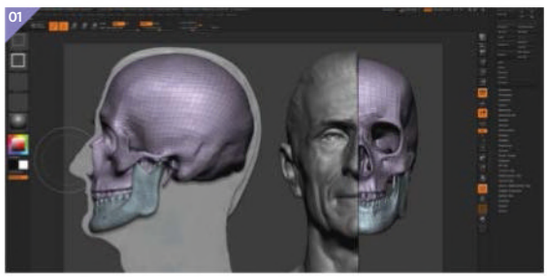
Concept
This piece is intended as a reference guide that looks at the critical anatomical forms of the face, including skeletal muscles, eye and nose construction, muscles of expression and fat deposits.

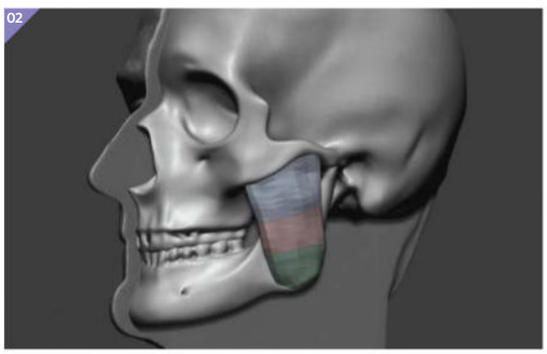
Master facial anatomy

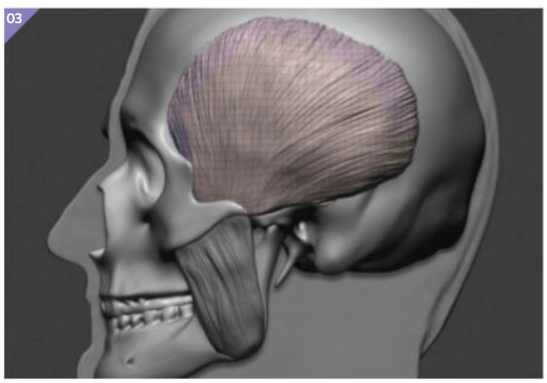
Model an écorché study and pick up the fundamental anatomical structures critical to portrait sculpting

This tutorial walks you through the steps of blocking in an écorché study of the face in ZBrush. Écorché is the age-old practice of sculpting or drawing the body without skin in order to study the forms of the underlying anatomy. Here we build up the main anatomical forms of the face and discuss their importance to portraiture.

Sculpting or drawing a face is one of the most challenging things an artist can attempt. Most artists wander in the dark when creating a portrait – placing planes and locating features based only on observation and experience. With lots of practice this can lead to good results, but everyone can benefit from a road map that helps them work faster and better. Knowledge of facial anatomy is just that – a guide to the forms and regions of the face. The face, like the body, is a puzzle of interconnected anatomical pieces. Once the artist learns common shapes and connections he can quickly block in a likeness. The goal of this tutorial is to familiarise you with these common anatomical structures.







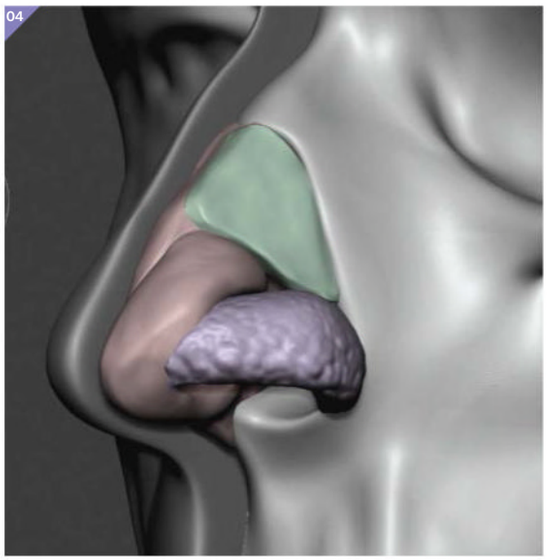
01 Fit the skull We first need to get the skull to fit the volume of the reference portrait supplied. Turn on transparency and adjust the proportions with a large Move brush. The areas of the skull that should be superficial (ie near the surface) are the eye orbits, the zygomatic arch (cheek bone), bridge of the nose and forehead. The rest of the skull has muscle, fat or cartilage between it and the surface. It is important to appreciate where the skull is closest to the surface so we can use these as landmarks when constructing a portrait.

02 Place the masseter The masseter is the first of two skeletal muscles that we need to place. Skeletal muscles connect bone to bone and articulate a joint (as opposed to facial muscles which articulate the skin to create expression). In this case we are connecting the mandible (jaw) to the zygomatic arch. The masseter creates significant surface form on the face and needs to be located correctly. In ZBrush, block in the shape with your preferred geometry creation method, then subdivide and detail.

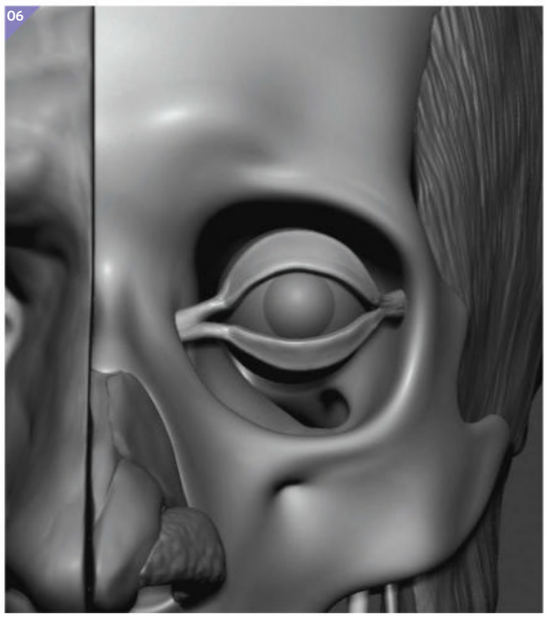
03 Understand the temporalis The temporalis is our second muscle of mastication (chewing). This muscle combined with the masseter, give humans a wider range of motion for biting and chewing. The temporalis is a large, distributed fan-shaped muscle that anchors to the skull along a crescent (the temporal line) that stretches from the upper-lateral corner of the orbit back to behind the ear. This distributed muscle body converges, focusing its power, and passes behind the zygomatic arch, grabbing onto the anterior point of the mandible called the coronoid process.

Geometry creation
In this tutorial we need to create a number of subtools for the various anatomical forms of the face. ZBrush gives us a smorgasbord of tools for creating geometry, so how you proceed is up to your own workflow and preferences. For this exercise I used a combination of ZSpheres with adaptive skin, the new QCube (under the Tool>Initialize menu), and InsertMultiMesh Curve brushes. My workflow involves appending a very low-res mesh that I can pull and shape with the Move brush, then I subdivide and detail using ClayTubes and alpha 54 or similar.

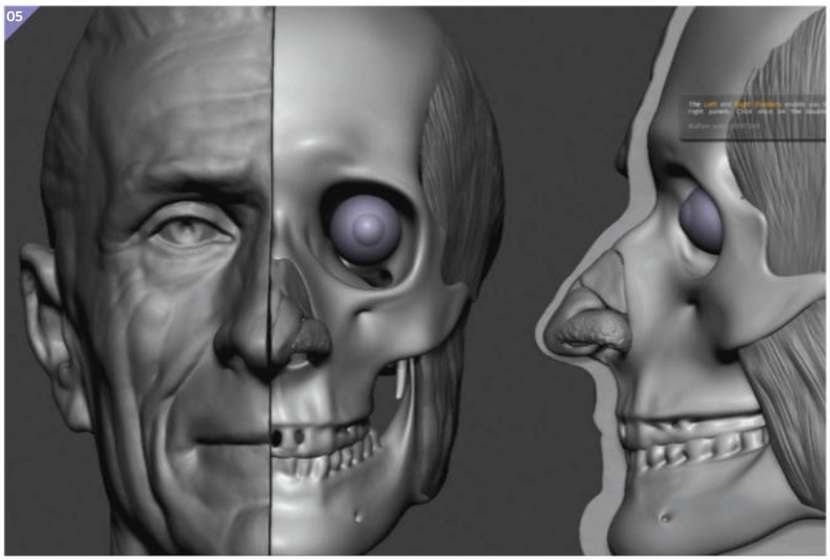
"The masseter creates significant surface form on the face and needs to be located correctly"



04 Construct the nose cartilage The nose exhibits an amazing amount of variation in life. It is important to understand that despite this variety, all noses are built from the same three pieces of cartilage and a small bit of fat. The septal cartilage establishes the midline of the nose. The complex shape of the alar cartilage creates the ball of the nose, and the lateral cartilage creates the side walls of the nose. Finally to funnel in the air we have a wedge of fat that creates the nostril of the nose. Using your best construction skills, construct this geometry over the nasal opening of the skull.



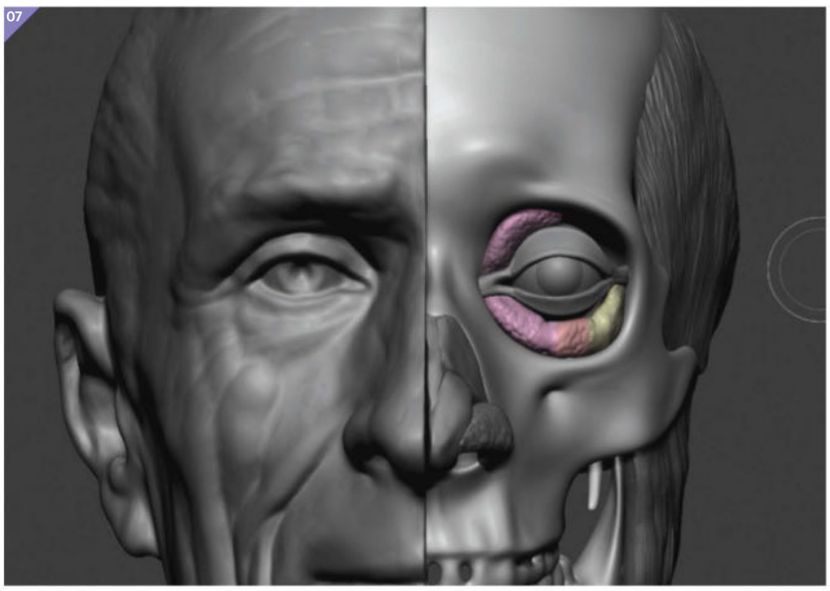
06 Make eyelids using tarsal plates The eyelids have two stiff plates of cartilage in them called tarsal plates. These form the foundation of the lids and are anchored to the orbits by something called the canthal ligaments. The medial (inner) canthal ligament is visible on the surface and so creates important surface form, the lateral one attached inside the rim of the orbit and is not visible. Paint a mask on the eyeball and then use Subtool>Extract to make your tarsal plates. Use your favourite geometry creation method to place the ligaments, then DynaMesh the result together.



05 Place the eyeball An adult human eyeball is approximately 24mm in diameter, there is small variation in this but it is a very useful average. The most common mistake in people's portraits (in CG anyways), is that they always make the eyeball too large, wedging a grapefruit into the orbit and then trying to get the eyelids to fit properly. Here, with a correctly sized sphere (do it visually as our skull is not to scale) centre the eye in the orbit, and move it forward to the point where the bone of the orbit still protects it.

Getting facial fat right

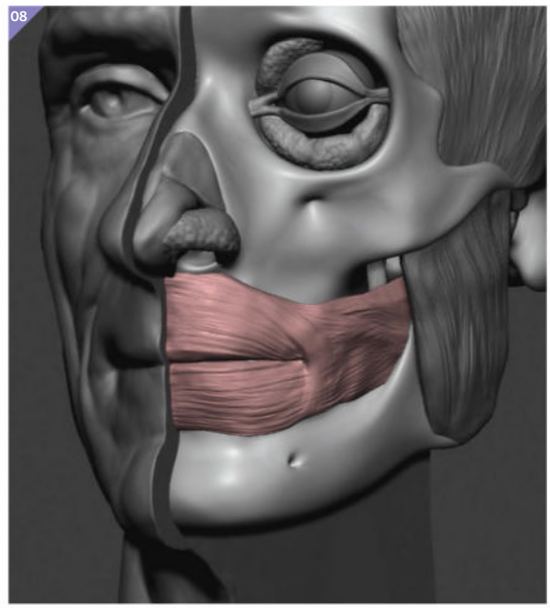
Not many artists appreciate how significant the fat deposits of the face are to the character of a portrait. In fact, almost every écorché study omits them. Fat is responsible both for the beauty and softness of youth as well as the craggy hollows of old age. In dissection, fat looks quite a bit like kernels of corn. To achieve a fatty texture in ZBrush use the Standard brush with Stroke set to DragRect and a suitably fatty alpha. I used alpha 62 with judicious use of the Smooth brush to texture the fat on this piece.



07 Construct orbital fat With the eye and lids in place you can see the large cavity we need to fill with orbital fat. Orbital fat is packed all the way into the orbit and is what cushions and holds our eyeball in place. With age we can see this fat pushing out of the orbit, giving us characteristic bags under our eyes. Construct half a donut of fat and locate it in the space between the tarsal plates and the rim of the orbit.

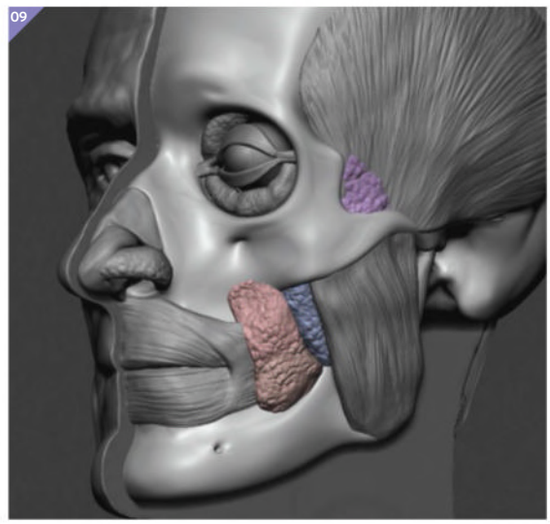
08 Put in the oribicularis oris

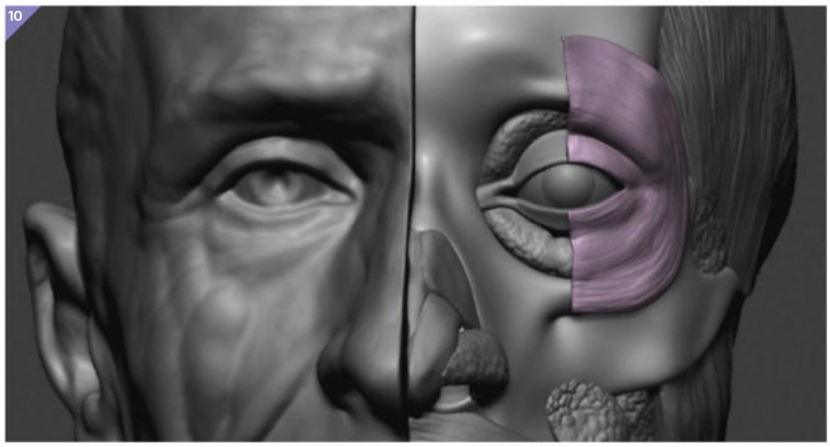
Our lip/mouth muscle is the first of two orbicular (circular) muscles on our face, made of concentric rings of muscle fibres, which pull inward in various combinations to make a variety of lip shapes and phonemes. The muscle is extremely complex as many surrounding muscles will pass fibres into it. Here we show the deep cheek muscle, the buccinator, merging into the orbicularis oris from the sides. The buccinator originates above and below the molars and passes fibres into the deep section of the orbicular muscle.



09 Make deep cheek fat pads

There are a couple of fat pads in the deep recesses of the skull that act as cushions for the facial muscles. These deep cheek fat pads sit over the buccinator and squish out into the depression in front of the zygomatic bone and a bit above the zygomatic arch. The volume of these fat pads determine the size of the hollow in a person's cheek. The man in this portrait has very hollow cheeks so keep the volume small as you construct these fat pads.





10 Cut orbicularis oculii The face's second orbicular muscle covers the orbit and eyelids and is responsible for blinking, squinting and raising the cheek. It extends quite far over the orbital rim and wraps medially to attach to the medial canthal ligament. Here we have cut away the medial half of the muscle to keep the orbital fat and tarsal plate exposed. Use ZSpheres or an IMM Curve brush to get geometry wrapping around the orbit, then flatten it and DynaMesh it. Use the TrimRectangle tool to cut off the medial half.

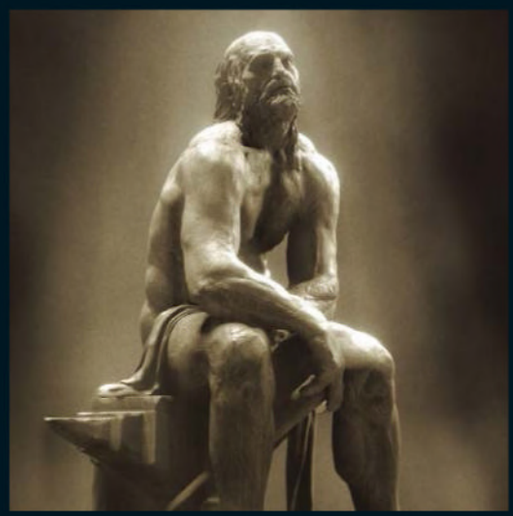
Showcase

Scott Eaton

Scott Eaton is an artist, character supervisor and anatomy instructor living in London, UK. He is a classically trained sculptor who has been teaching anatomy to digital artists for over a decade. Scott also contributes his expertise to film and commercial VFX and he has worked as lead modeller and character supervisor on numerous films. He has taught anatomy to studios around the world including Industrial Light & Magic, Valve, Framestore, Ubisoft and many others.

Torso Studies **ZBrush, Mental Ray (2014)**
Study of the articulation of the shoulders and back from Scott's digital figure sculpture online course.

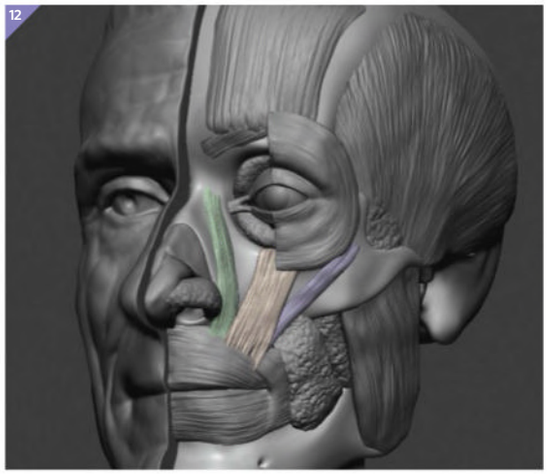
Mermaid Study **ZBrush, Mental Ray (2011)**
Unused mermaid concept sculpt from *Pirates Of The Caribbean.*



Hephaestus **ZBrush, Mental Ray (2009)**
Digital sculpture of the Greek god Hephaestus. Hephaestus was the crippled blacksmith of Mount Olympus, shown here in repose at his forge.

11 Build forehead muscles – temporalis and corrugator These two muscles are important contributors to the expressions of surprise, anger and sadness. They are thin muscles so they don't create surface form but we certainly recognise the expressions they create. For the temporalis, create a flat sheet of geometry from just above the orbit to the top of the forehead. Medially it stretches almost to the centre, laterally to the temporal line. The corrugator is a small diagonal muscle that spans the space between the base of the nose to a point approximately mid-eyebrow. Place this using a small piece of geometry.

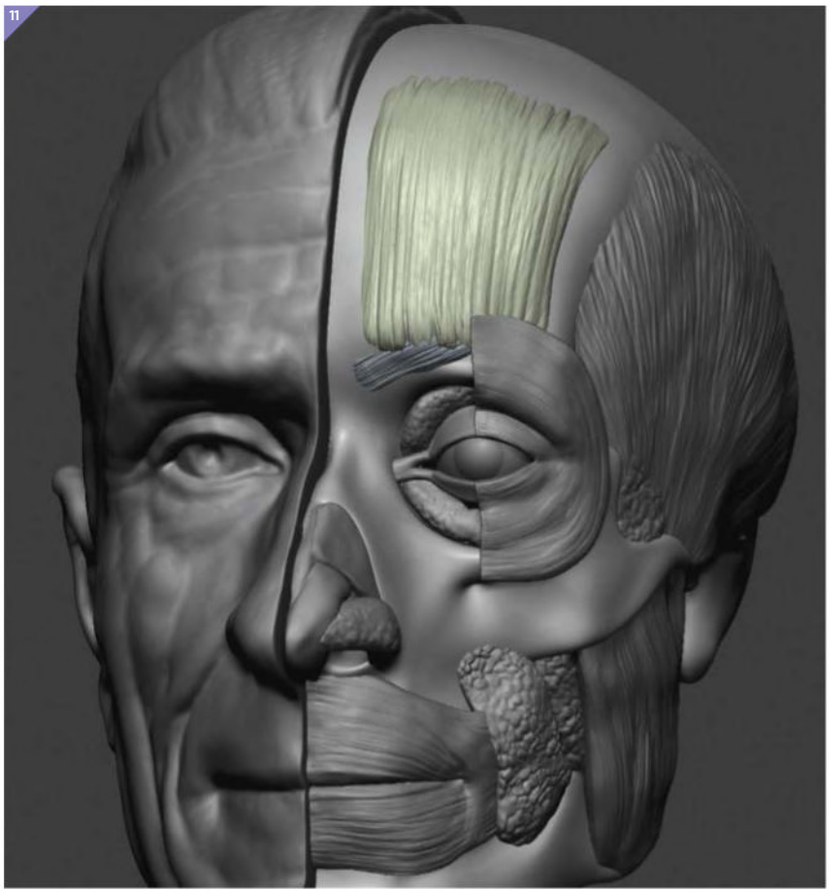
12 Lift the lip The human face has a network of muscles that articulate the upper lip. The muscles anchor into the upper edge of the orbicularis oris muscle (as well as intermingling fibres into the lip). They are called (from left to right): levator labii superioris alaque nasi, levator labii superioris and zygomatic minor. They are respectively responsible for wrinkling our nose, sneering and subtly deepening the nasolabial furrow (the furrow that comes off the nose and proceeds down by the corner of the mouth). Construct these like little pieces of linguini. Note: there is also a sneaky corner-lifting muscle not shown here.

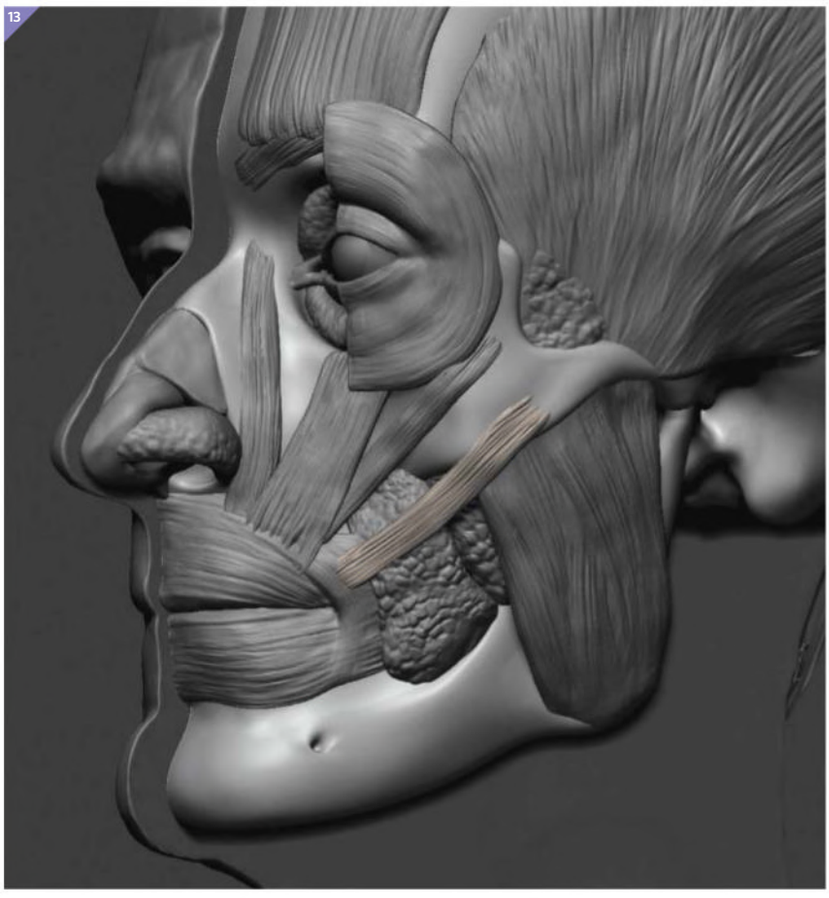


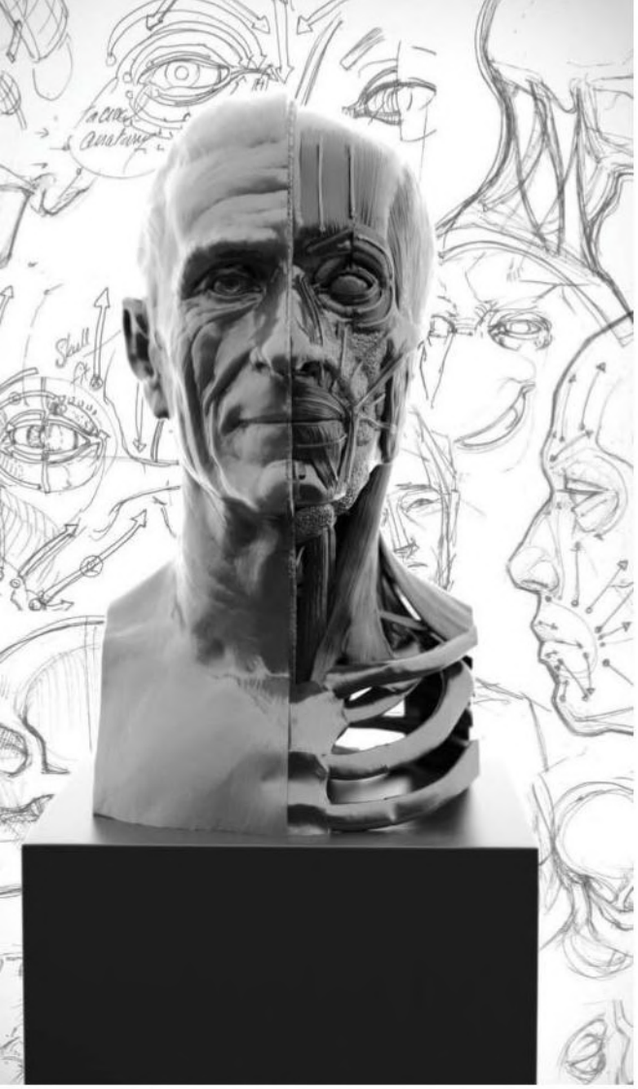
13 Construct the zygomatic major The muscle, zygomatic major, is our smile muscle. It can be difficult drawing or sculpting a smile but once you understand the origin/insertion and action of this muscle, the construction of the smile becomes easy. Build this muscle from the corner of the mouth out to the lateral surface of the zygomatic arch. It has the widest pull of any of our facial muscles and lifts and stretches the lips when it activates, creating a big smile.

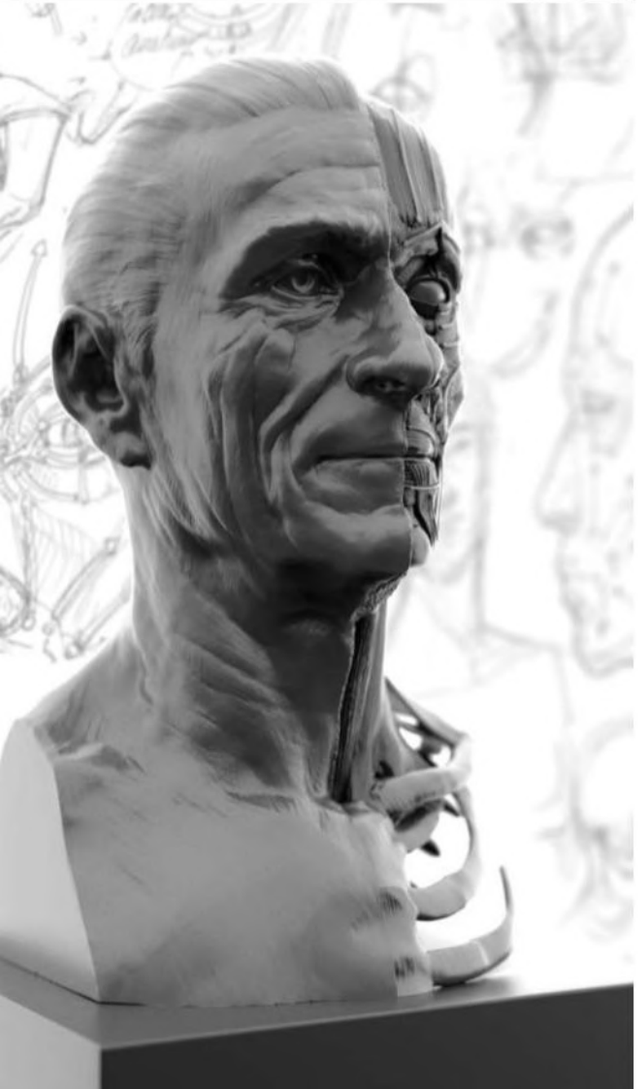
Facial muscles

Facial muscles often create very little surface form so you might think they aren't interesting to us as artists, but we absolutely need to know them because of the expressions they make. Once we know the construction of the muscles, ie the origins and insertions, we can derive the expression it creates. Each muscle also has characteristic wrinkle patterns that we are programmed to read, even when very subtle. There is a lot to learn about expressions but the best way to start is by studying the underlying musculature.





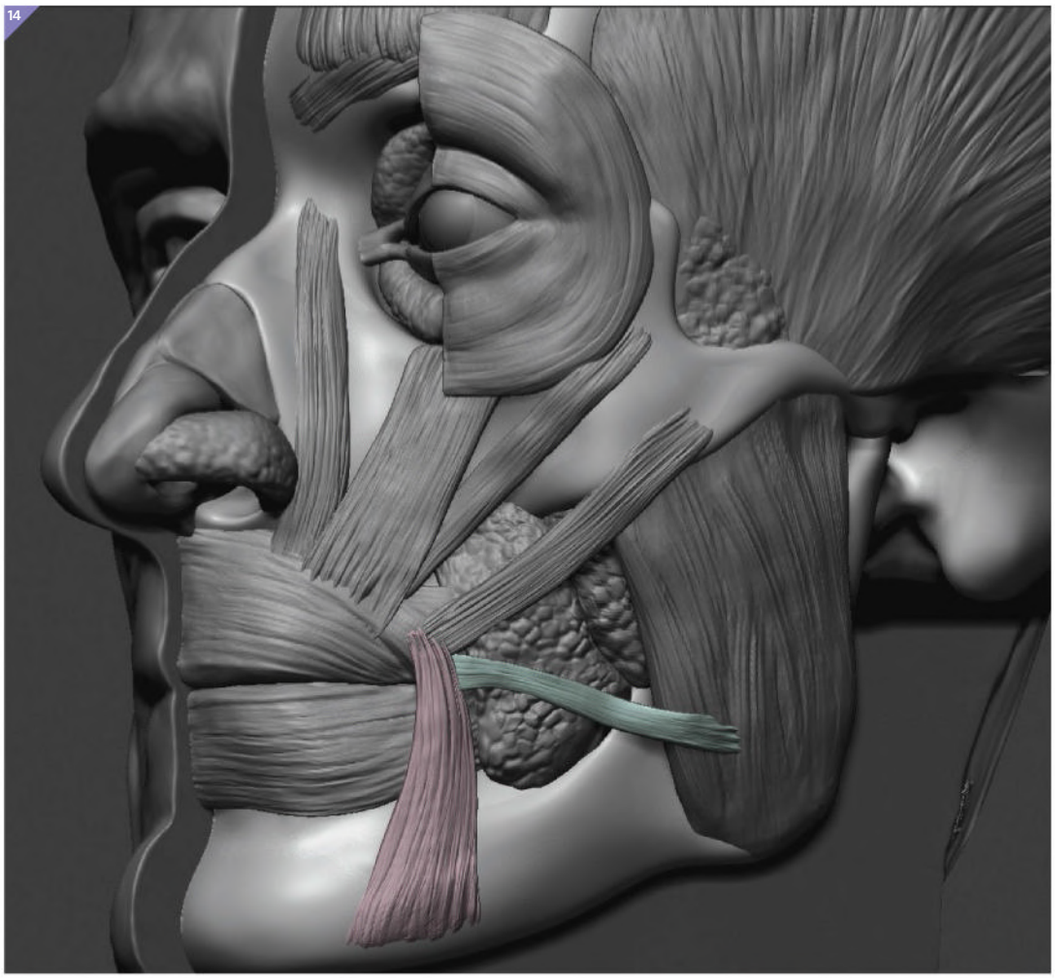


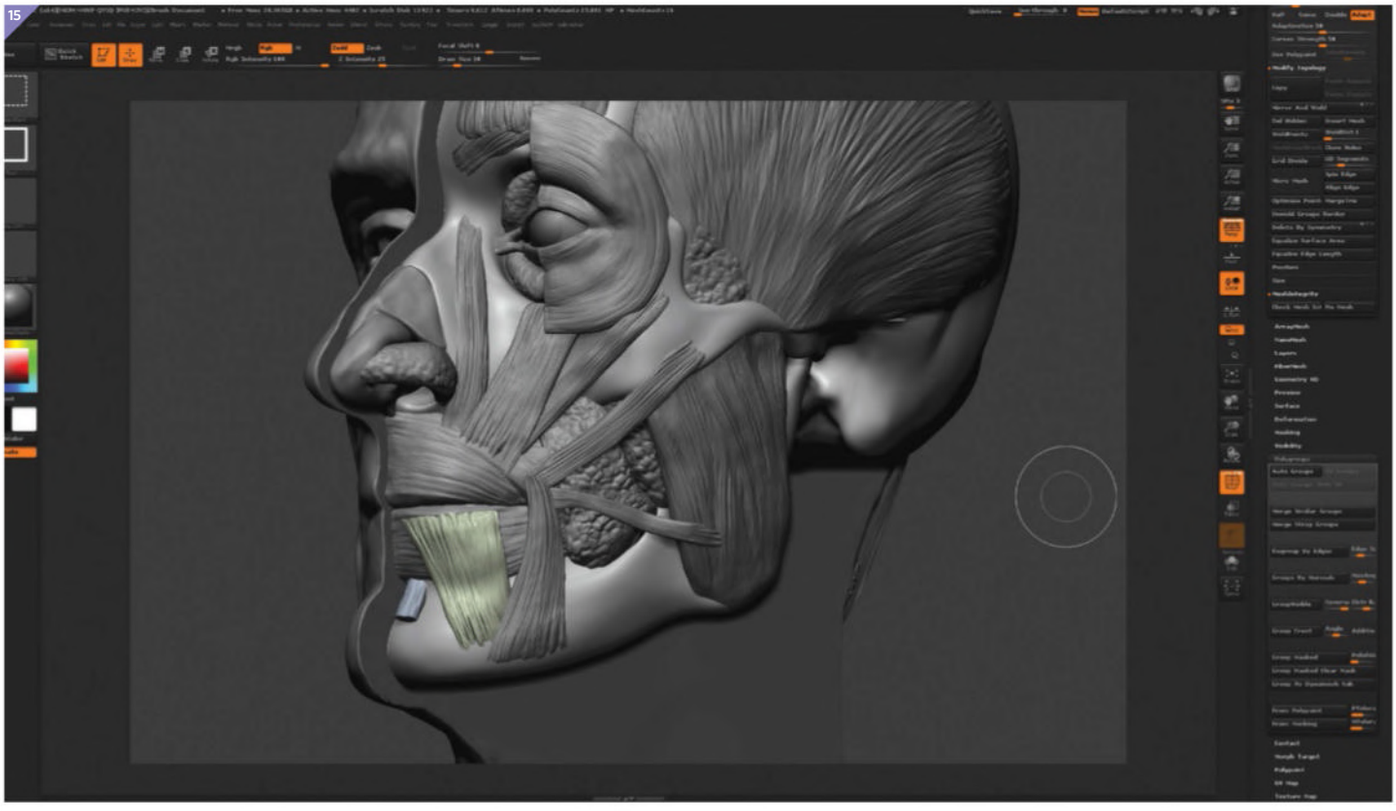


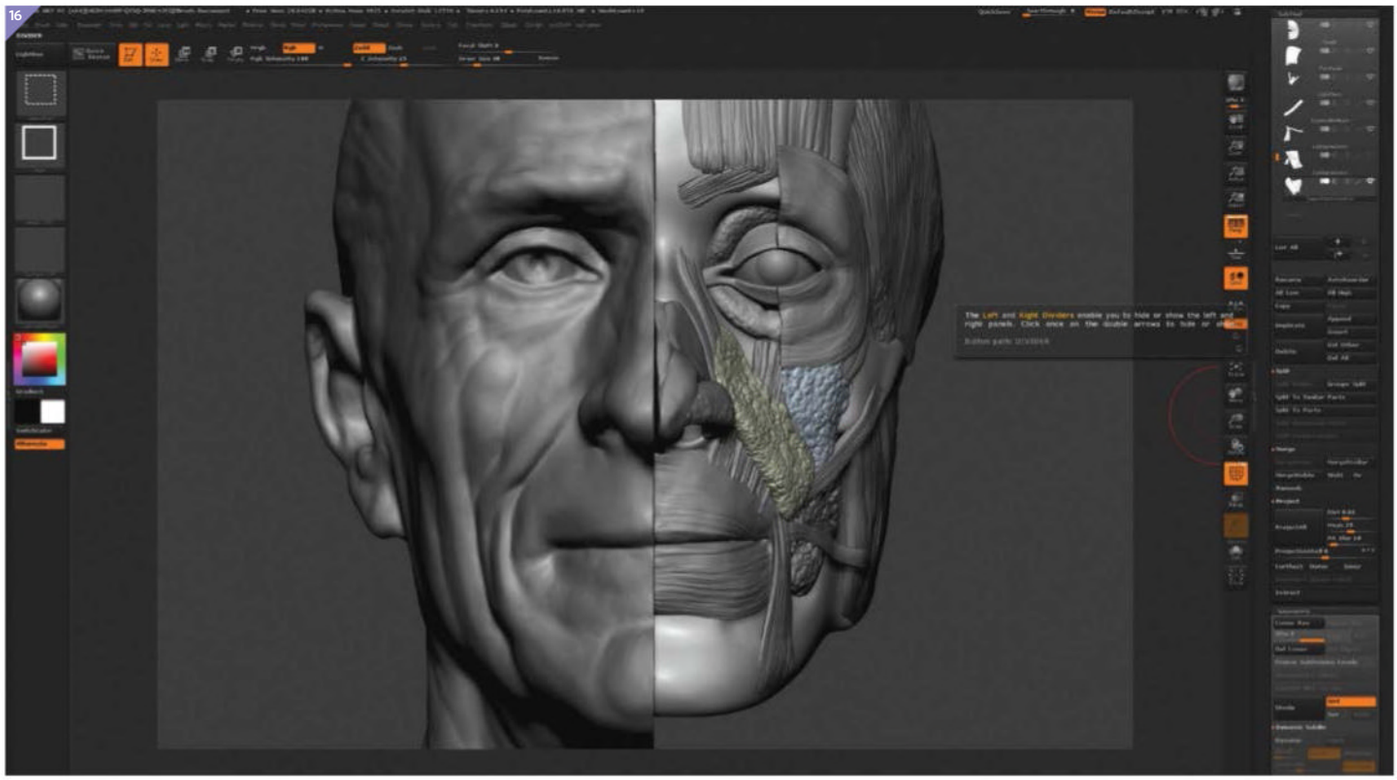
For the temporalis, create a flat sheet of geometry from just above the orbit to the top of the forehead. Medially it stretches almost to the centre, laterally to the temporal line

14 Pull down corners Just as we had zygomatic major pulling the corner of the mouth upwards to create a smile, we have a corner depressor called depressor angulii oris, that pulls the corner down to create a frown. This muscle pulls on the corner of the mouth but also passes fibres into the upper part of orbicularis oris. Some of this form can be visible on people's faces. A second small, but related, risorius muscle pulls the mouth wide and slightly downward. Unusually, it actually originates from the skin over the masseter.

15 Depress the lip and lift the chin The opposite of the upper lip lifters (Step 12) is the lower lip depressor, depressor labii inferioris. It anchors onto the jaw just inside the depressor anguli (Step 14) and then ascends, grabbing onto most of the body of the lower lip. This flat sheet of muscle retracts your lower lip showing your bottom teeth. Adjacent to this muscle is the last muscle of the tutorial: the mentalis. This funny little muscle anchors between the lower teeth and chin, and grabs onto the fat pad that covers the chin, raising it when activated. It is shown cut away here.



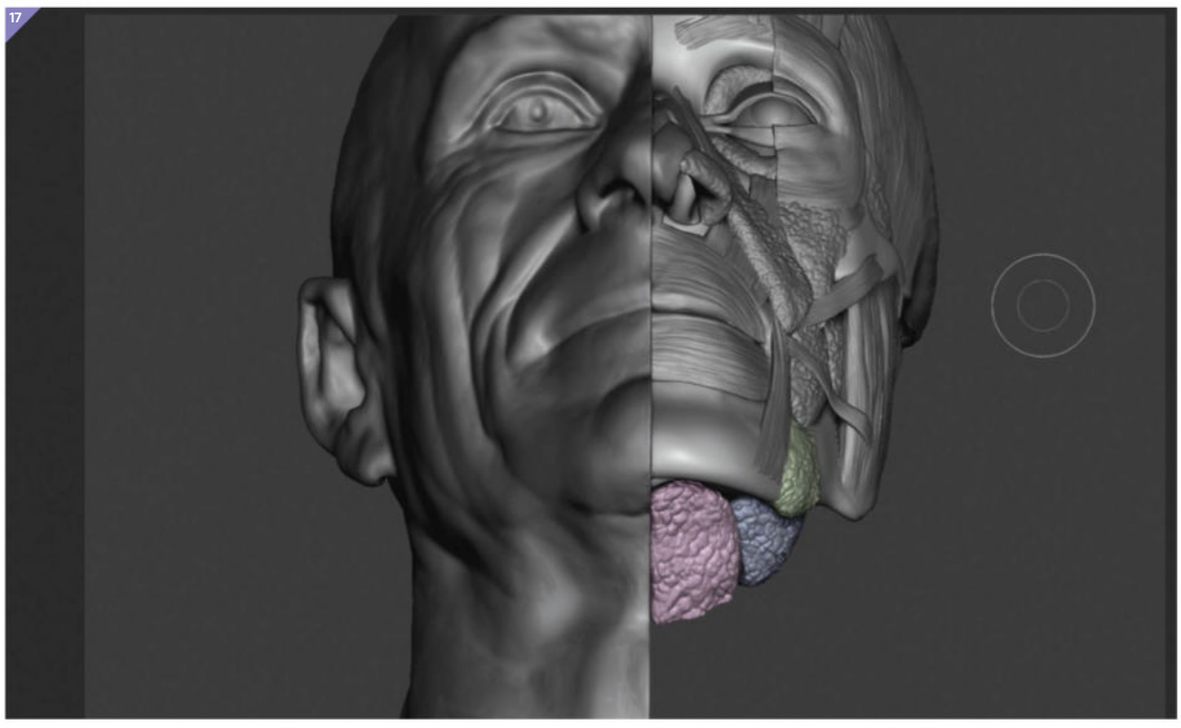




16 Construct the malar and midcheek fat Now that the muscles of expression are in place we need to start covering them with subcutaneous fat. We have a critical fat pad, called the malar fat pad, that is located below the bottom of the orbit and lateral to the base of the nose. The malar fat pad helps create the nasolabial furrow. As we age it sags and separates from the other subcutaneous fat deposits. Here we also construct the triangular mid-cheek fat pad, also quite defined on our model.

Final thoughts and continuing study

We have come quite far in a 17-step tutorial. I tried to fill it with useful anatomical information but space is short. I hope this tutorial has excited your interest in facial anatomy and that you can appreciate how knowledge of facial structure can improve your portraiture. There is so much more information to cover on all fronts but we will leave that for another time. If you would like to learn more, I refer you to my website (**scott-eaton.com**) where I have a number of additional resources to help you continue your study.



17 Build the submental fat and jowls The last few characteristic facial fat deposits are located on the lower part of the face and under the chin. The submental fat compartment, our double chin fat, is constructed underneath the chin. Try to be careful not to extend it out too far to the sides, nor too low either. Adjacent to this are a pair of jowl fat compartments, one above and one below the jawline. You should construct these like small parcels of fat that gravity has had its way with over time.

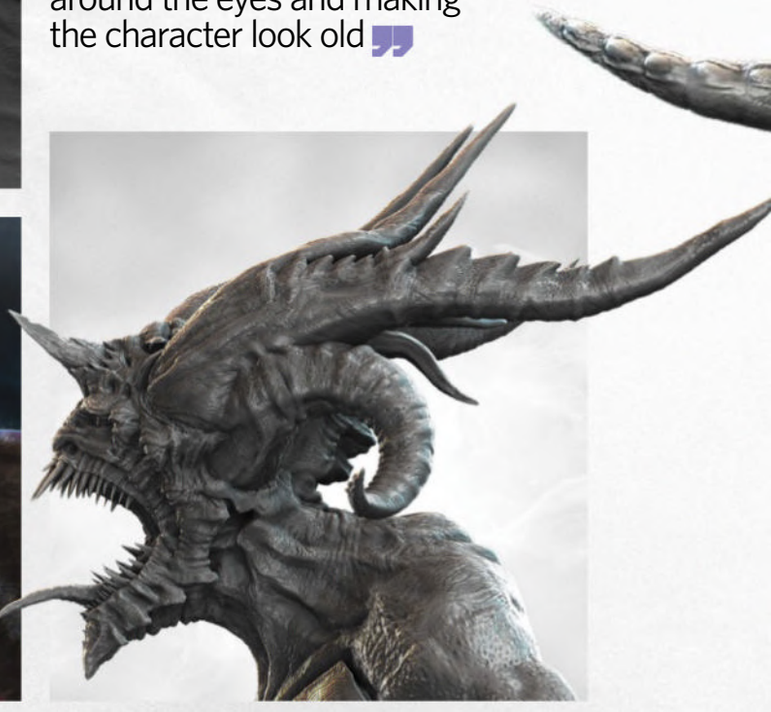
Sculpt a bloodsucking vampire

Discover how to develop, pose and use render passes to make this master of vampires in ZBrush

Here we will create a posed and detailed character bust. Starting from just a 3D sphere, we will rough out the bust and then sculpt the organic forms and features. We will also make use of the readily available Insert Mesh feature to build the character. Later we will move on to editing these and then resculpting them so that they fit in to the design. After posing the bust we will use polypainting to create basic materials and textures for the model. Finally, we will take the 3D renders into Photoshop to finish the composition by adding lighting effects and textures.

Print your own *Master* bust using the file provided on filesilo.co.uk

> Dam_Standard is great for enhancing the facial expression by drawing fine skin folds around the eyes and making the character look old



JAMES SURET
The Master, 2014

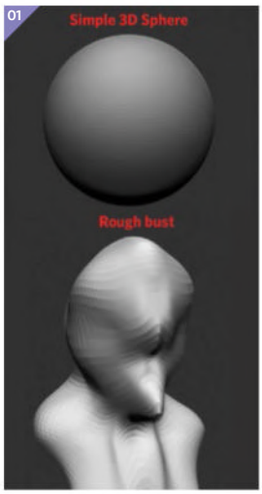
Software
ZBrush, Photoshop

Learn how to
- Create basic anatomy from just a 3D sphere
- Use Insert Mesh to quickly build your character
- Pose the character using masking and Transform tools
- Use DynaMesh to rebuild the topology of your mesh
- Use polypainting to paint materials and textures directly on to the model
- Create multiple render passes to create a high quality final image

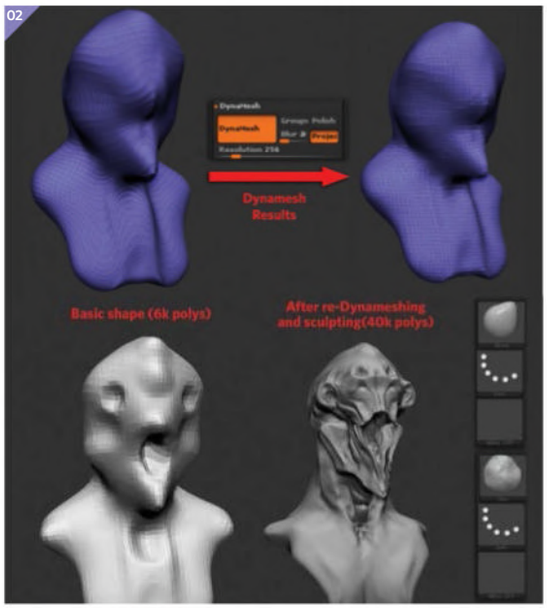
Concept
The aim for this project for me is to create a detailed and expressive character bust in a horror style, and make the character as realistic as it can possibly be.

YOUR FREE DOWNLOADS
from **filesilo.co.uk/bks-798**
- Tutorial screenshots
- Printable STL file

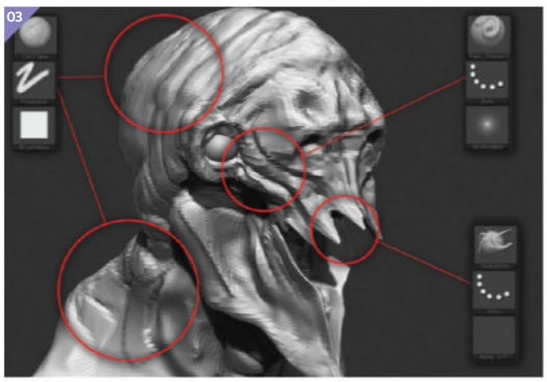
01 Create a rough bust First we insert a new 3D Sphere, press T to edit the object, click Make Polymesh3D and activate Symmetry by pressing X. Now with the Move brush set to a large brush size (by pressing the] key), we can push the shape around. The idea is to create the shape of a character bust loosely based on human anatomy. Initially, the important parts to work on include pulling out the nose and chin, and then pushing in the neck.

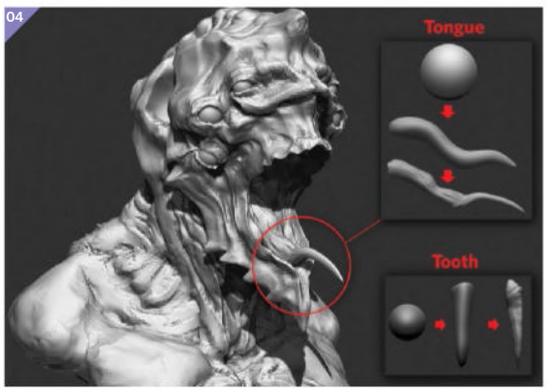


02 Develop the model using DynaMesh Now we have a rough form, but the mesh is becoming difficult to sculpt because of its topology. So use the DynaMesh feature with the resolution set to 256 – this creates a more even mesh to work with. We can then create more of the basic features. When we feel we need more polygons to add more detail, use DynaMesh again but with a larger resolution. At this point the poly count is quite high so start using the Clay brush to carve in smaller features such as the eye sockets, lips and muscles.

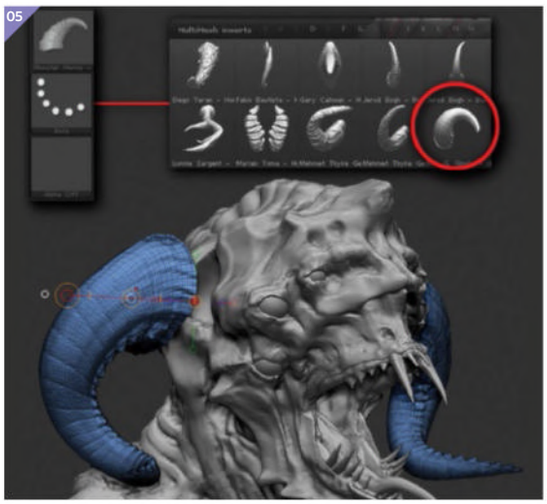


03 Sculpt the initial details The next step is to add character and realism to the model by sculpting basic skin folds and muscle structure. Using the ClayTubes brush we are able to quickly create the look of muscle fibre and bone. The Dam_Standard brush is great for carving lines and wrinkles. The SnakeHook brush is useful for quickly pulling out sharp shapes such as spikes and teeth. We are exploring ideas for this character's features and we may resculpt or replace them later. Next, we add the eyes by inserting a sphere. Mirror the sphere using the SubTool Master plugin, then move the eyes into place with the Transform tools.



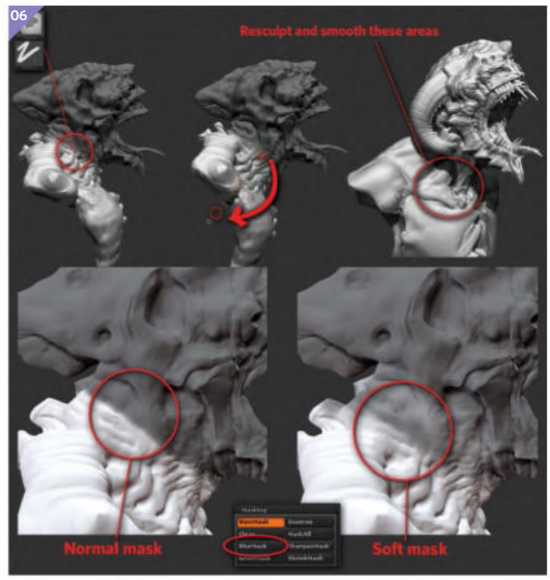


04 Add facial features Mask out the lower jaw by holding the Cmd/Ctrl key and painting the mask on, then hold Cmd/Ctrl and click off the model to invert the selection. Using the Transpose tool, rotate the unmasked area by pressing R and move the end of the transpose line. To create the teeth insert a sphere. Pull out the shape of a tooth with the Move brush, use DynaMesh to even it and sharpen using the SnakeHook brush. Use the same process for the tongue and run the ClayTubes brush over the tongue for an organic look.

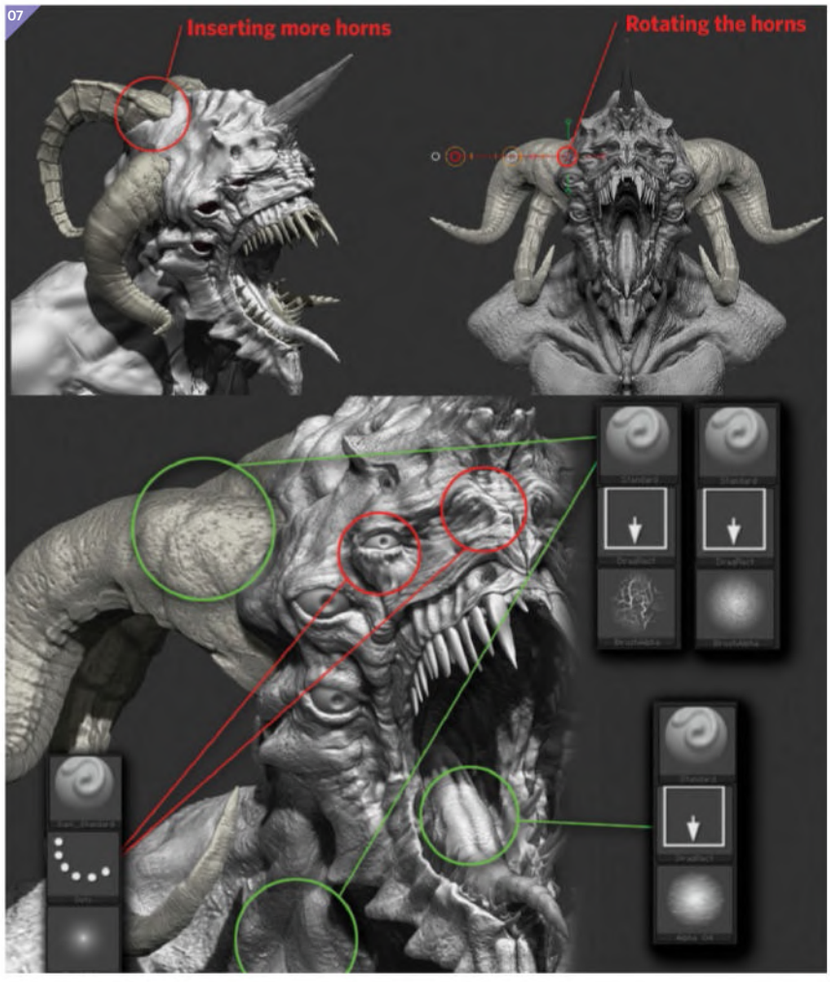


05 Design with Insert Mesh Now that one tooth is sculpted fill the mouth with teeth. Using the Move tool draw out a Transpose line, hold down Cmd/Ctrl and drag the centre circle out to duplicate the object. Then resize the tooth with the Transpose tool. Next, add horns to the character. Speed this up with a premade horn from the Monster Horns and Antlers brush pack available for free at www.badking.com.au. Click on the side of the character's head and drag out the desired horn size. Then use Transpose to position it.

06 Pose the monster Using the Masking tool, mask off everything from the middle of the neck downwards, then holding Cmd/Ctrl, click off the model to invert the selection. Using the Transpose tool now would create a distorted area around the edges of the mask. So to help solve this, hold Cmd/Ctrl and click a few times inside the masked area to soften the mask. Then when you rotate the area it will create a smoother bend. Either way we will need to resculpt these areas after posing, but this step will make all that much easier.



07 Change features and details To develop the aggressive look, add another set of horns from the brush pack and rotate both sets of horns to almost intertwine. To balance the features, pull out the central horn and sharpen it using the Move brush. Use DynaMesh again at a higher resolution to sculpt more surface details. Now use Alpha materials with the Standard brush to stencil surface detail onto the model. ZBrush comes with some Alpha materials but there are also free ones at pixologic.com/zbrush/downloadcenter. With Alpha selected and Stroke set to DragRect we can quickly add surface details to the skin and horns.



Showcase

James Suret

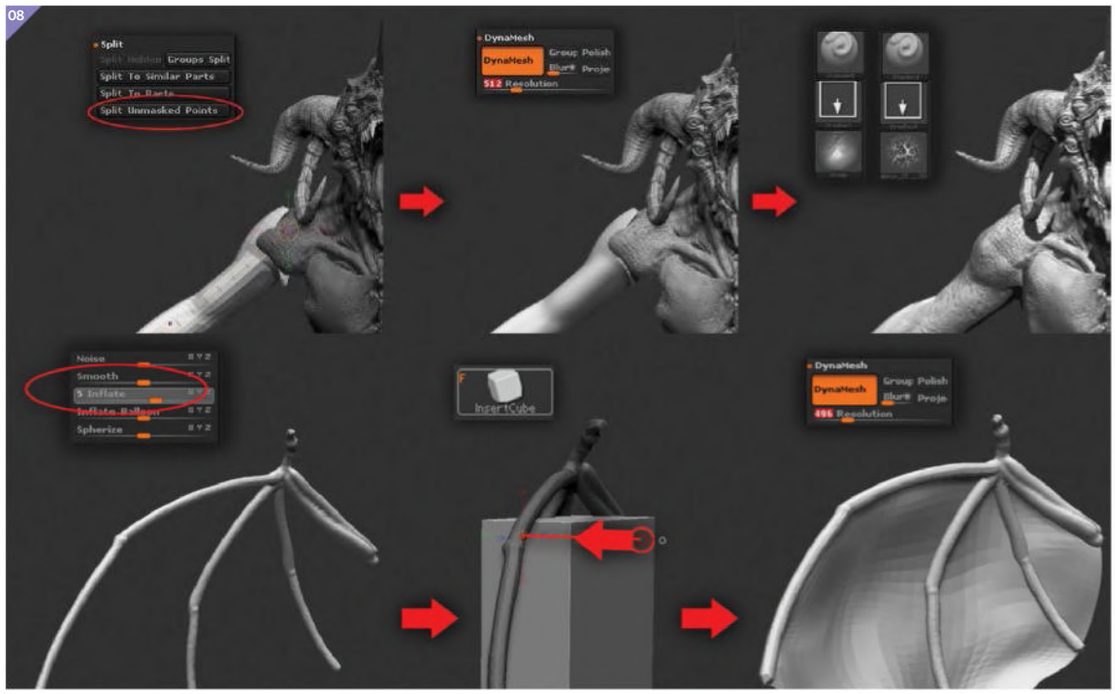
I am currently a full-time web developer and freelance 3D artist. I am constantly developing my portfolio with the aim of working as a concept artist in the games or entertainment industry. My main focus is creating original creatures and illustrations.

The Maw of the Sea **ZBrush, Photoshop (2014)**
The last stand against a giant monster. This piece was sculpted in ZBrush, and the paint and effects were done in Photoshop.

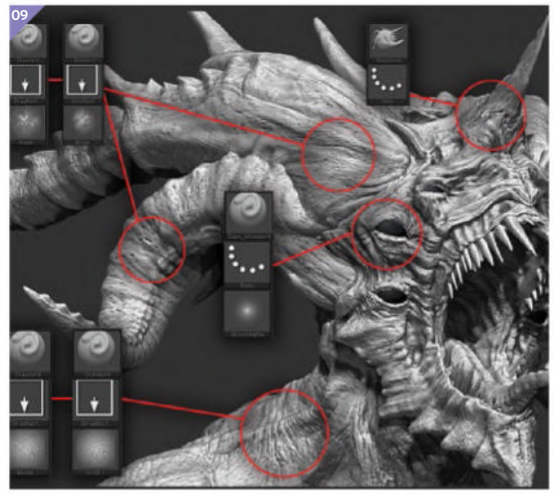
Victory **ZBrush, Photoshop (2014)**
A female cyborg displays her war banner on the battlefield. The cyborg was sculpted in ZBrush as a digital maquette.

Alien Cyborg **ZBrush, Photoshop (2013)**
A 3D character concept of an alien cyborg. This was sculpted in ZBrush with composition and effects done in Photoshop.

08 Add the arms and wings The arms are inserted from the human body parts Insert Mesh set that comes with ZBrush. Clicking on 'Split Unmasked Points' under the SubTool menu will create a new subtool. Use DynaMesh on the arms to smooth them out and increase the poly count. Now use Alpha materials to add skin details. The wing structure is inserted from the Dragon Insert Mesh set. Use the Inflate function under Deformations to flesh out the skeleton. Then insert a cube and resize its depth using the Transform tools. Now we can pull the cube out into a wing shape using the Move brush and even the surface with DynaMesh.

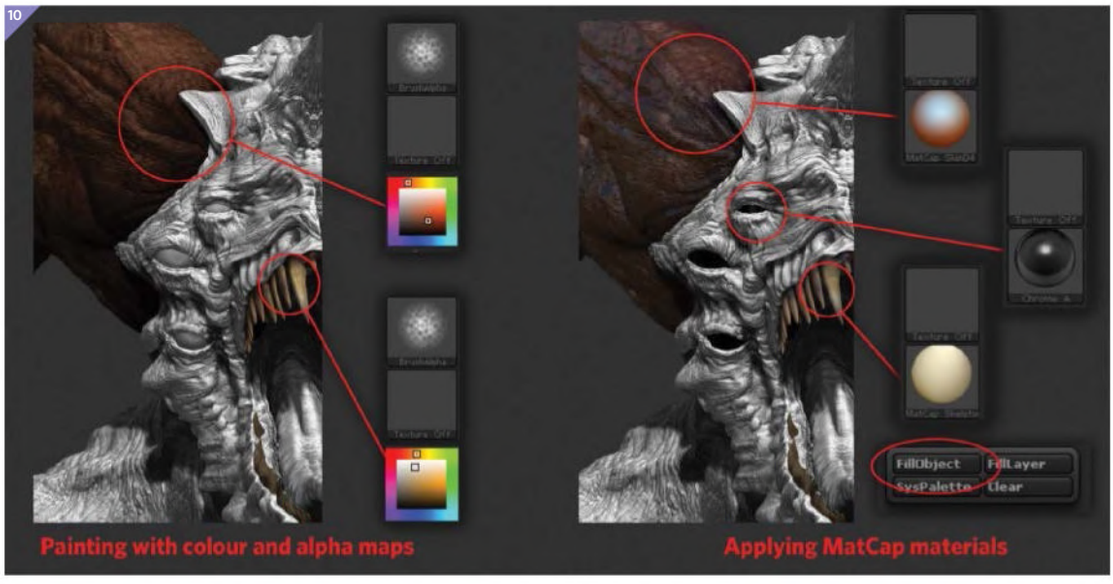


09 Final details Now we are happy with the overall design of the head but it looks too large. To remedy this add more horns and join them together. The idea is to give the character an ancient look. Now finish off the sculpt by adding surface details and small features. Use the SnakeHook brush to pull out spikes and bone shards on the top of the head. The Dam_Standard brush is great for enhancing the facial expression by drawing fine skin folds around the eyes and making the character look old. Then use several Alpha materials to layer hard and sharp textures onto the bones and soft bumpy textures onto the skin.



Alternative ways to start a character

We started this model with a simple sphere and shaped it into a rough human form. You can also use the human bust model DemoHead.ZTL that comes with ZBrush for sculpting directly, or you can insert a sphere, make it the same head size and activate Transparency mode. The latter method will let you rough out the shape of your model whilst being able to use the DemoHead bust underneath as an anatomical reference.



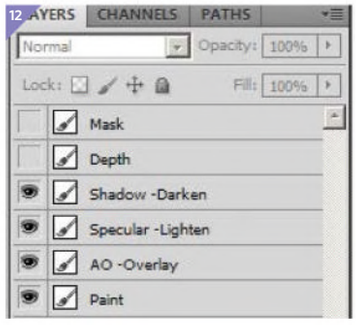
10 Polypaint the base textures Fill the model with a plain material by clicking Mrgb and selecting MatCap White01 as the material and FillObject in the Color menu. Make sure Zadd and Zsub aren't ticked (so we don't sculpt on the mesh), and tick RGB as this will enable us to just paint with colour. It's usually a good idea to choose an Alpha material to paint with so that we can blend colours naturally. Then apply some materials to enhance the textures. Make sure only M is ticked, select a material and click FillObject from the Color menu.

11 Lighting and rendering First we need to resize the document. For a landscape orientation we set the width to 3,500 pixels. Rotate the model to create an interesting composition bearing in mind the Rule of Thirds. Next, move the light placement by moving the dot around the sphere under the Light menu. Now set the shadow resolution to 8,000 on the BPR Shadow section of the Render menu to create higher quality shadows. Then enable Ambient Occlusion under 'Render properties' and set its resolution to 8,000. Click BPR render and save all of the render passes. Finally, fill the model with ToyPlastic material and black colour. Then click BPR render to create a specular layer.

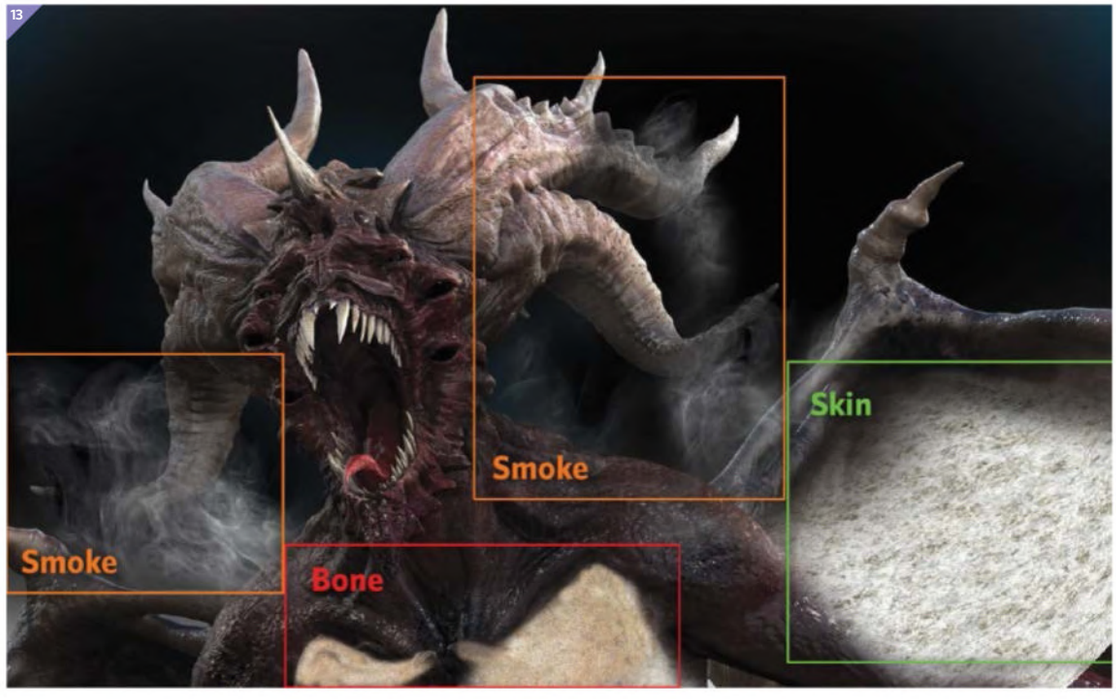
Kitbashing with Insert Mesh

Kitbashing is a method used to quickly add large features or fine detail to your models. ZBrush comes with a large collection of premade model parts. These can be inserted into your sculpture and duplicated and resized with Transpose. Once inserted, these objects can be resculpted just like the rest of your model. This lets you create a quick 3D concept of your character and later you can change the features to meet your design brief.

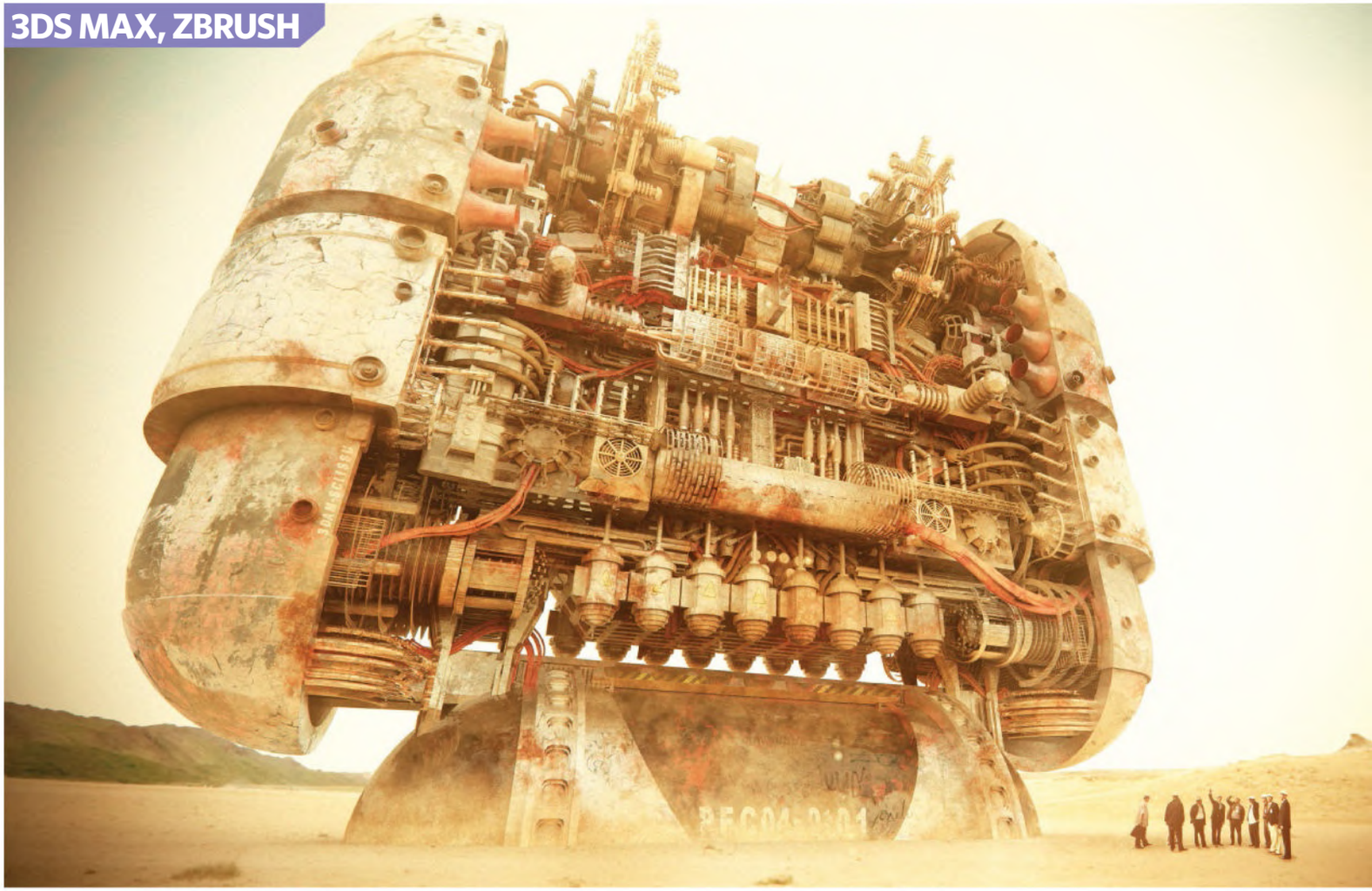
12 Composite the layers in Photoshop After saving all of the renders as PSD files, open them in Photoshop and layer them up. First the main BPR render, followed by the AO layer set to Multiply – it helps add more depth to the render. Next, set the specular layer to Lighten, which brings out highlights along the skin. Finally, set the shadow layer to Darken to enhance the dark areas. The mask layer can be used to delete the background of the renders to enable us to place a background image behind the character.



13 Add effects and textures It is always a good idea to show a character concept in some kind of environment to help bring it to life. To do this, add a background stock photo and apply a strong Gaussian blur to create a depth-of-field effect. Next, insert some stock photos of skin and bone to enhance the textures on the character; set them to Overlay and reduce the opacity. A great selection of free photo textures can be found at cgtextures.com, among others. Finally, with the help of a soft brush we will add some soft lighting and glow effects to the teeth, eyes and bone areas. This emulates reflected light and adds to the atmosphere of the image.



3DS MAX, ZBRUSH



3ds Max, ZBrush

Pablo Castaño
facebook.com/eletecedateando

Pablo is a concept artist, 3D/VFX designer and musician working in TV, videogames and film

Create sci-fi details in 3ds Max and ZBrush

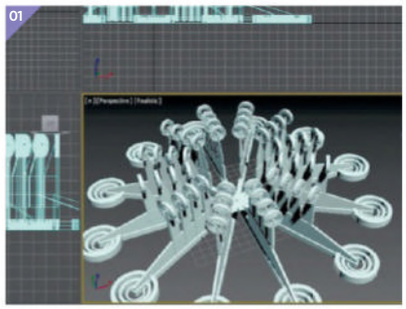
In this tutorial we are going to tackle the task of creating a huge amount of details in our models in a really easy and fast way. At first glance, the idea of designing a complex and detailed model (a mech for example) can be pretty scary. By building small pieces to make one larger piece every step of the process turns into a design challenge rather than a technical one. Take, for example, the interior of a desktop computer which has an incredible amount of details. If you look closely, you'll notice that every piece is composed of a large number of basic shapes: cubes, cylinders, cones, planes and so on. This is by design, we as humans cannot create complex structures from thin air, we need to start from the basics, from the small and simple to the large and complex. Nature uses the same principle but through fractal repetition of growing patterns. This process of modelling an intricate 3D model using simple pieces can be really fun and relaxing when you are using kitbashing techniques in ZBrush.

First we are going to focus on creating a toolkit of low-poly pieces in 3ds Max, then we are going to prepare our objects for exporting as an OBJ format. Once we have it all imported in ZBrush we are going to split every piece into individual SubTools in order to create an Insert Multi Mesh brush (IMM). This can be a one-time process or an incremental one, once you have your IMM you can use it in a future sculpt. The piece that you planned to use as a surface detail in an electrical circuit can be used tomorrow as part of a futuristic skyscraper concept.

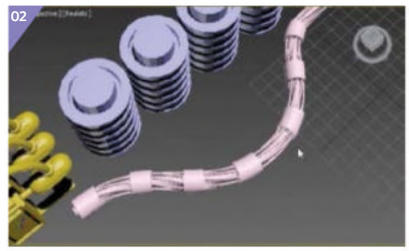
If you really want to develop a unique form and come out with a style of your own, take maybe 15 or 20 minutes a day, open your software of choice and model a few pieces. Try different styles, look for references online and play with the forms. Try to kitbash by yourself, take one object and merge it with another, change the length, the width and the scale. Your imagination is the limit. It's really not important which software you use to create the kitbash pieces, this tutorial uses 3ds Max but you can use the one you like the most – just keep in mind the poly count, you don't want to crash ZBrush in a couple of strokes!

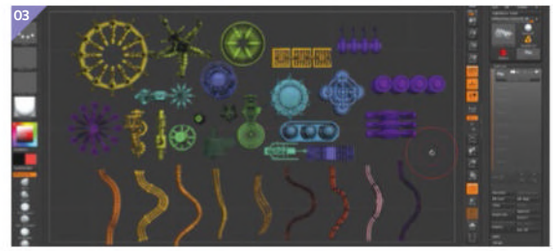
YOUR FREE DOWNLOADS
from **filesilo.co.uk/bks-798**
- Tutorial screenshots
- Video tutorial

01 Build the objects in 3ds Max First create a few basic forms inside 3ds Max. Create a box on the grid and subdivide the height, length and width twice. Then go to Modifiers>Edit Poly, pick a few polygons, and use bevel and extrude to mess with the profile of the box. If you need to create a nice circular component for your concept try the circular array function in 3ds Max. Let's say that we had created an object made from a couple of boxes, a cylinder and a torus. Select all of the pieces, go to the top menu then go to Group->Group...>Hierarchy>Affect Pivot Only and move the pivot far away from the grouped object (this will be the centre of the circular array) and click on the rotation tool (if you skip this step the pivot is not going to be fixed for rotation translations). Now, in the top menu go to Tools>Array in the Rotation option. Click on the right arrow and this will enable the absolute values for the array angle. If you have the top view selected, enter 360 in the Angle text box for the z axis and press Preview. Now you can increment or decrement the number of copies of your array in the Count text box. Other cool details for a hard surface model are cables and pipes. Create a cylinder in the viewport, it has to be long and thin. Then go to Modifiers>Edit Poly-polygon select a few polygons and extrude it inwards. Again, mess with the shapes by using bevel, inset or extrude – just have fun. Once you are happy with the cable it's time to bend it. Create a spline (Create>Shapes>Line), click and drag to draw a curve, and guide the deformation of the cable. Select the cylinder, go to Modify>Quadify Mesh, choose Modify>Path Deform(WSM)>Pick Path then select the spline. If the result is not what you expected click 'Move to Path'.

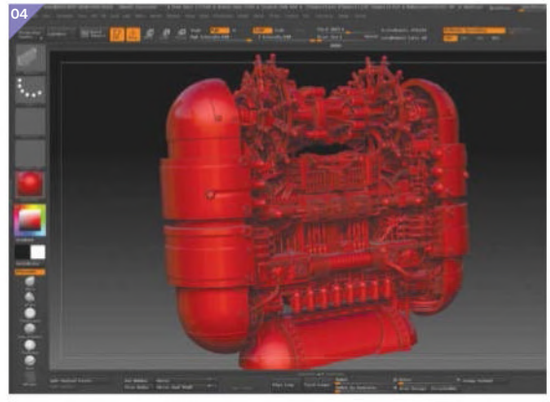


02 Export the pieces from 3ds Max Once you think you have enough pieces ready, order every one of them side by side as this is an important step for creating the IMM: every piece has to face the same direction. Select each piece (and all of its constitutive objects) and collapse it (Utilities>Collapse) so we can be sure that every piece is a unique object. It's time to export! So go to File>Export and select OBJ:Exporter in the menu.



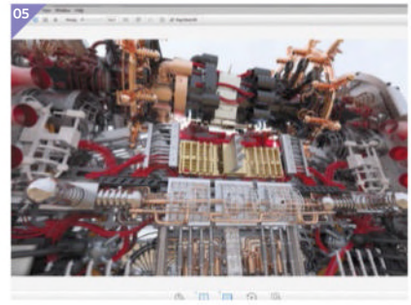


03 Create the IMM in ZBrush In ZBrush create an object in the canvas (a cylinder, a cube, or any other shape – it doesn't matter), enter Edit mode, click on Import in the Tool menu and select the OBJ file that you exported earlier from 3ds Max. All of the objects from 3ds Max should now be on screen, if you can't see them all just simply press F to frame all of the geometry in the viewer. But there is a problem! We only have one SubTool! Don't worry, if you press Shift+F you will see that every object is a different Polygroup. Use Opt/Alt+click or a pen tablet to align the objects flat on the canvas, the bottom must point away from you and the head towards you. This is important because the IMM will instance the geometry according to the orientation given by the SubTools in the moment of the creation of the brush. Go to SubTools>Split>Group Split and this will separate every object into a different SubTool. In the top menu click on Brush>Create InsertMultiMesh. If you look in the brush palette (press B to get to it) you will find your IMM brush at the bottom. Click on 'Save as' and save it in your BrushPresets folder (C:\Program Files (x86)\Pixologic\ZBrush 4R7\ZStartup\BrushPresets). Your IMM brush is now part of your brush palette.

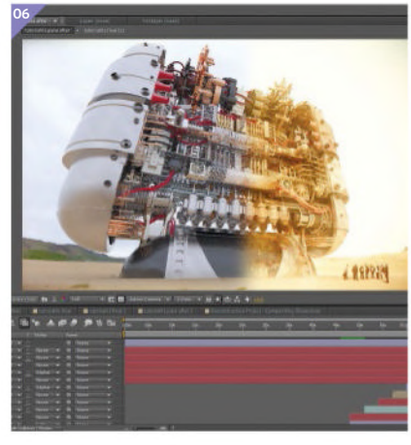


04 Build a base structure and kitbash it Let's face it, we can't start from the details, we have to have a structure to support our little objects in the first place. If you plan to add details to a model that you already have then skip this step, but if you are starting from scratch you will need a larger object to place the tiny pieces. Just put a box in your scene, DynaMesh it and cut away a few chunks with the ClipCurve brush, or use ZModeler to come out with a SubTool that enables detailing work. Once you feel happy with your base structure load your kitbash brush and press M on the keyboard, select a piece and brush it on the surface of the object that you have just created. Remember that you can use the symmetry option of ZBrush. Every time you do a stroke, a new detail will be created. If you want to separate every piece in a new SubTool press 'Split Masked Points' under the SubTool>Split panel after each new piece. This is the basic procedure for creating a custom IMM and using it in ZBrush, the rest is texturing, render and postproduction.

05 Materials and render On this occasion we will be using KeyShot for the render. In ZBrush go to the Render tab and enable the external renderer (ZBrush 4R7 and KeyShot Bridge). In KeyShot assign the materials to each SubTool by dragging and dropping the material from the library. Choose an environment and render the image, Remember to enable the Z-Depth pass inside the Render dialog in Keyshot.



06 Postproduction For the composing and tweaking of the render passes we are going to use Adobe After Effects. First, make a new comp and drop the base render and the Z-Depth pass in the timeline. Hide the Z-Depth pass and in a new adjustment layer add your favourite depth-of-field plugin (Camera Lens Blur or Frischluft's Lenscare to name a couple), select the Z pass inside the plugin and play with the parameters until you achieve the look that you are looking for. With Curves, adjust the RGB contrast as needed. You can also throw a grungy texture or fog footage below the adjustment layer and try a few blending modes to give a more weary look to the final composition.



When you're happy with the overall shape you can start using the Slash brush to define the different plates, seams and cuts

JONATHAN BENAÏNOUS
Sci-Fi Helmet, 2015

Software

ZBrush, Maya, Photoshop

Learn how to

- Sketch your concept directly in ZBrush
- Create nice and clean topology in Maya
- Generate Alphas in ZBrush, using Maya and Photoshop
- Detail your hard surface model at an expert level
- Render your different PBR and lighting passes in ZBrush
- Generate a final picture using render passes in Photoshop

Concept

I aimed to create a hyperrealistic and well-polished sci-fi helmet with a distinctive style, drawing reference from a number of sci-fi inspirations.

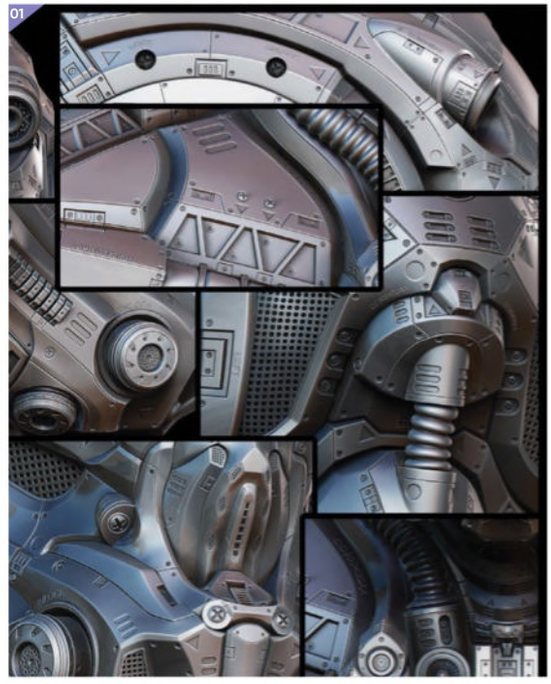
YOUR FREE DOWNLOADS
from **filesilo.co.uk/bks-798**
- Tutorial screenshots

Design a hard surface helmet in ZBrush

Discover and learn the entire creation process of a very high detailed sci-fi helmet, from the ZBrush concept to the final picture

Over the next few pages we will explain how to go through the different steps of production of a very detailed hard surface helmet such as this one. At face value, the final image may seem discouraging to some readers as it exhibits a high level of detail and polish. However, in this tutorial, we will attempt to extract the method, techniques and tricks that will assist you in reaching this high level of render quality. It is fundamental for you to be aware of your own capacity as an artist, and to be very critical of your own work at the same time. The more you demand of your own work and the harder you push yourself, the more you will end up developing your skills. We will also learn how to take advantage of powerful pieces of software such as ZBrush, Maya and Photoshop. We will show you how to sketch your concept directly in ZBrush, and how to use Maya to clean and refine your ZBrush model. Finally you will also learn how to use your PBR render and ZBrush to make a final composited image in Photoshop.

01 Find inspiration The first step is to find a number of references to help inspire your creation. This is a very important step in the process of producing an asset such as this. The references you find will aid you in determining the direction of your style. Take a good amount of time to analyse all of your references in detail, to absorb and identify the elements you would like to have on your future model (eyes, ears, jaw, sensors and so on). This step is the foundation of your project and the most crucial part. Your work here is going to directly impact your final result, so make sure you are happy with your defined style and have found adequate references before progressing.



02 Sketch the helmet Using a basic head as a guide will give you a great starting point. Append a sphere around the head, then once converted as DynaMesh, start searching for interesting shapes using the Move brush. Use the ClipCurve brush to get sharp slices and curves on your silhouette. When you're happy with the overall shape you can start using the Slash brush to define the different plates, seams and cuts on your model. Use the TrimDynamic and the hPolish to clean the different surfaces. Try to keep from having too much blobbiness in the mesh and to keep enough definition between the individual pieces.

03 Sketch the neck and shoulders

Once you are satisfied with your resulting sketch, use the Lasso selection tool to isolate pieces and split them into different SubTools. To give a thickness to each piece, be sure to have one single Polygroup by SubTool. Then, in the Deformation subpalette use the Polish By Groups slider to clean the border of each piece. Next click on Close Holes, and you should now have a full piece. Select the inside Polygroup and extrude it using the Transpose tool to create the desired thickness. Now you should begin working on the neck and shoulder. You can use an IMM brush to insert tubes and pipes if necessary.

04 Refine all of the pieces

Use the TrimDynamic, the hPolish and also the ClipCurve tool again to clean and refine the SubTools as much as possible. Make sure you spend some time cleaning the thickness and the inside, assuring they have sharp and clean edges. Adjust the SubTools with the Transpose tool to create nice rhythm and variation between the pieces. Try to alternate the height of plates, make sure to readjust the connection between all the shapes using the Move brush if necessary. Remember to use this brush carefully and with a larger radius to avoid deterioration in your refining work.


04


03

Open or closed version

In our case we are currently creating a functional helmet but in a closed version. Of course, even if the model is going to stay closed, it is very important to make it believable. You should remember to keep some space between the face and the helmet if you are planning on creating an inside or an opened version. Use the transparency combined with Ghost mode to adjust the helmet with greater precision. It's also very important to hide and unhide some SubTools to check if the eyes, nose, mouth and ears are in the right place.

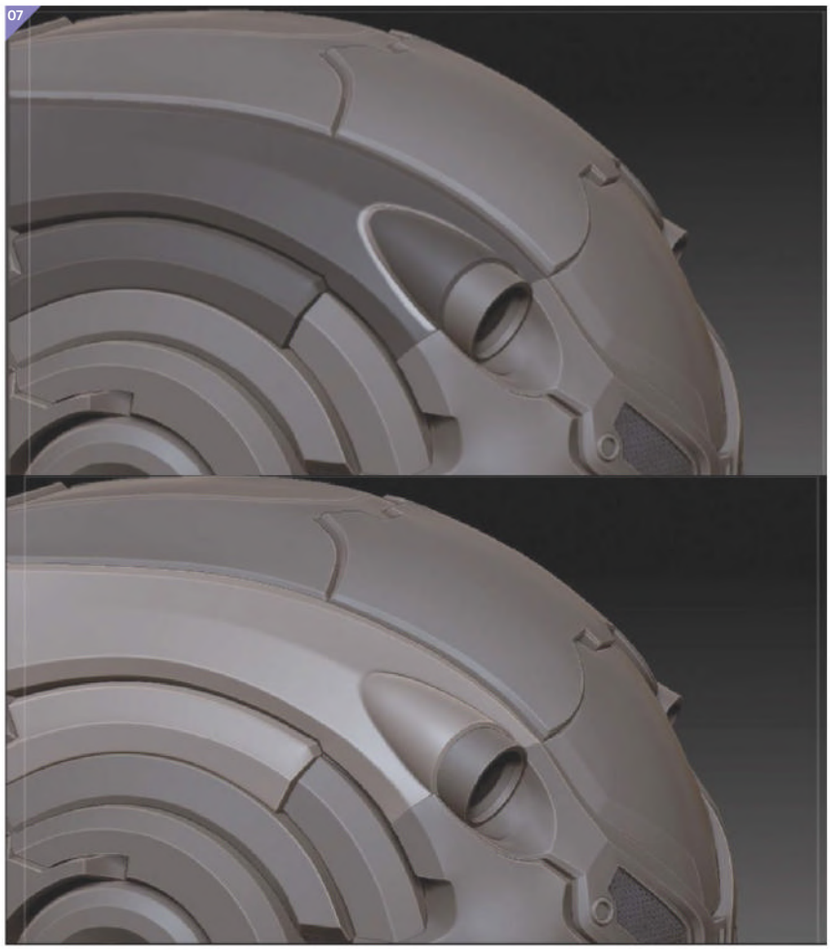
05 Retopo in Maya

In the Zscript menu use Decimation Master to decimate each of the SubTools. Try to have a good balance between the shape definition and the poly count to be able to import your model in Maya. Once in Maya you can use the Quad Draw tool to retopologise each element. Try to keep your topology as homogeneous as possible. In this manner you will have the same definition in ZBrush as when you increase the levels of subdivision. Use the Crease tool on each edge that you want to keep as a hard edge. This will generate a bevel on your creased edges. Previsualise the result using keys 1 and 3 to fix any issues properly.

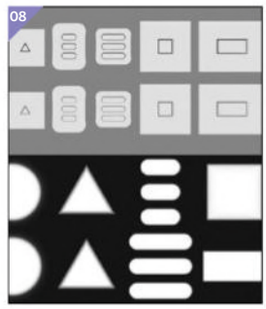
06 Transfer with GoZ

Use GoZ to transfer each SubTool from Maya to ZBrush. We are using this method because it keeps your crease information (that we got from Maya) for subdividing SubTools without losing hard edges. This process lets you find the perfect balance between sharp and soft edges. You can also play with the Crease Level slider in ZBrush to see if you want to keep the crease information or not when subdividing. Ensure you correctly merge all your vertices in Maya to avoid holes being created when you subdivide.


06



07 DynaMesh and merge your elements Before converting each SubTool in DynaMesh, start to delete the lower level of subdivision by clicking on the Del Lower button in the Subdivision tab. Use the Merge button to combine the SubTools you want to fuse and click Cmd/Ctrl+W to create one single Polygroup. Convert your SubTools in DynaMesh and clean your seams, if necessary, by using the Smooth brush. You can also mask the seams and play with the Polish slider in the Deformation subpalette to smooth your intersections in a uniform way to get a seamless result. Don't forget to use the Smooth brush and the hPolish to attenuate some pinch and tension on your surfaces.



08 Create Alphas using Maya and ZBrush In Maya, begin modelling the shapes you would like to use as Alphas. In our case, we are creating triangles, squares, rectangles, circles and cylinders (or capsules). You can have two different versions – one sharp and one that is much smoother, or one that is flatter and another that is deeper. Don't forget to keep a flat surface around your shape to avoid creating any cuts when you convert your shape into an Alpha in ZBrush. Now import your Maya geometry into ZBrush and use the GrabDoc to capture and convert each shape into an Alpha.

09 Create text Alphas using Photoshop In Photoshop, create a 1K squared new document and fill the page with solid black. Then type, in a white colour, some different text with a serial number and if you wish, a logo for the model. You can create a straight and a curved version of the text. As you can see in our screenshot, each text piece is a separate 1K texture. The more angle variations you have, the easier it will be to place your text along curved shapes.

Showcase

Jonathan Benaïnous

I started in the videogame industry almost ten years ago as a 3D artist and environment artist. I currently work at Ubisoft as a senior environment artist. I have a passion for creating a wide range of images, vehicles, environments and lighting.

Condenser Room / Beyond: Two Souls **Maya, Photoshop (2014)**
For *Beyond: Two Souls* I was in charge of creating this huge Condenser Room, including the modelling, the texturing and also the lighting.

Helicopter / Beyond: Two Souls **ZBrush, Maya, Photoshop (2013)**
For *Beyond: Two Souls* I've created this game asset inspired by the famous Black Hawk.

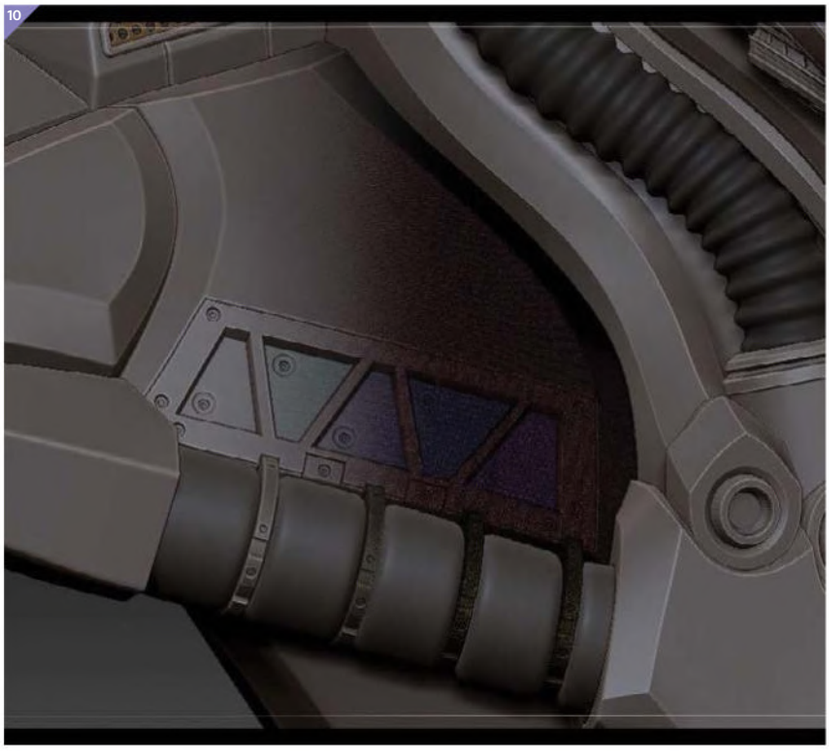
Eva 01 / Neon Genesis Evangelion Tribute **ZBrush, Maya, Photoshop, DDO (2014)**
One of my personal projects, this is a tribute to the animated Japanese show *Neon Genesis Evangelion*.

Tips for hard edges

To get the desired result when you are subdividing in ZBrush, play around with the Crease Level slider. Start subdividing four or five times at level 15 to maintain very sharp edges. Then just subdivide again two or three times at level 0 to break the sharpness and this will result in some well-bevelled edges. This trick will enable you to easily adapt the general aspect of your model. It is an essential step towards creating round and soft or really sharp and neat elements.

10 Detail in ZBrush

To add important details on your surfaces and create a great industrial pattern in ZBrush, use the SliceCurve to slice your SubTools into different Polygroups. Once the outline of your shapes are drawn, use the Transpose tool to extrude, scale and translate your Polygroups. Try to stay consistent with your different pattern and be sure to reuse some of them here and there to ensure that it's homogeneous. Use the Curve Frame Mesh button to generate a spline around your Polygroup's border, then use the CurveMultiTube brush to create nice rubber joints around your shapes.


10


11

11 First and second detail pass To start your first detail pass, use the Standard brush with a small draw size and use a lazy mouse to draw and delimit different plates on your model. The goal here is to underline the existing shapes and add nice lines of detail. Once you're happy with this, start using your custom Alphas from Maya to detail your pieces (switch the focal shift to -100 to avoid any distortion when you apply your Alphas). We suggest implementing the second and first pass together to keep a good balance between lines and patterns. Attempt to have logical positioning of your details to accentuate the realism; referring to previously gathered references here will help.


12

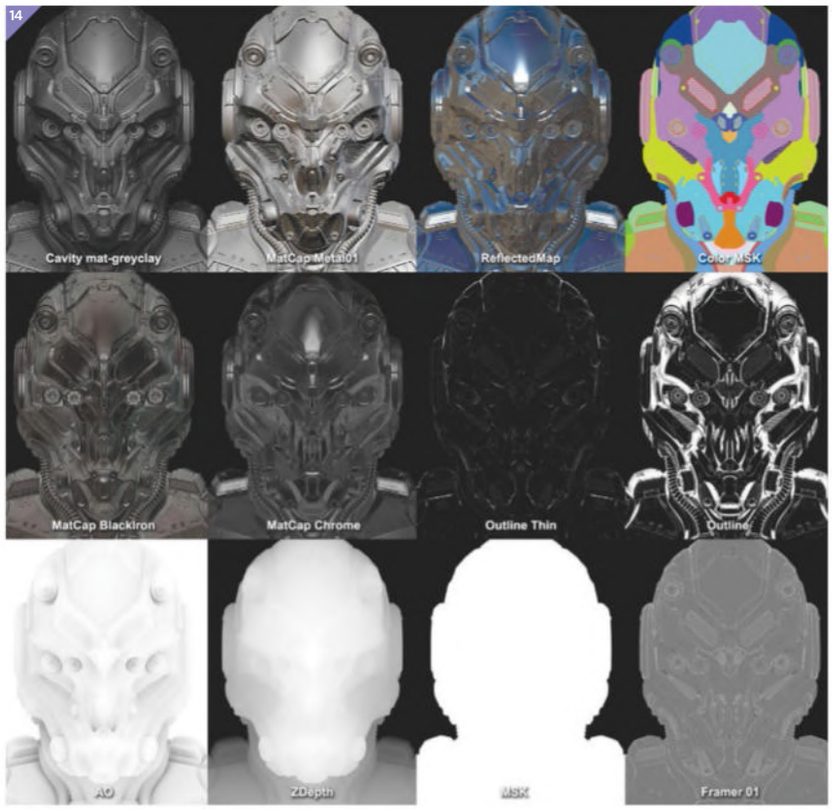
12 Third detail pass To start this detail pass, disable the symmetry and start using your text Alphas from Photoshop. Use the Drag Rectangle stroke to find interesting places to add your text. Once again don't forget to turn the focal shift to -100. Try to create a different reading level using different sizes of typography, for example smaller texts can be used as serial number on your different pieces while bigger pieces could be used as information for the user of the helmet. Try to imagine someone actually using the helmet and the information they would require.

“Play with [Decimation Master]’s percentage slider to optimise your SubTools and find a good balance between performance and accuracy”

13 Optimisation for faster renders Use the Decimation Master plugin to decimate all of your SubTools. Then play with the percentage slider to optimise your SubTools and find a good balance between performance and accuracy. Note that you should not notice any differences from the original mesh. Don't be too aggressive with the poly count and spend time testing different settings for a perfect balance.

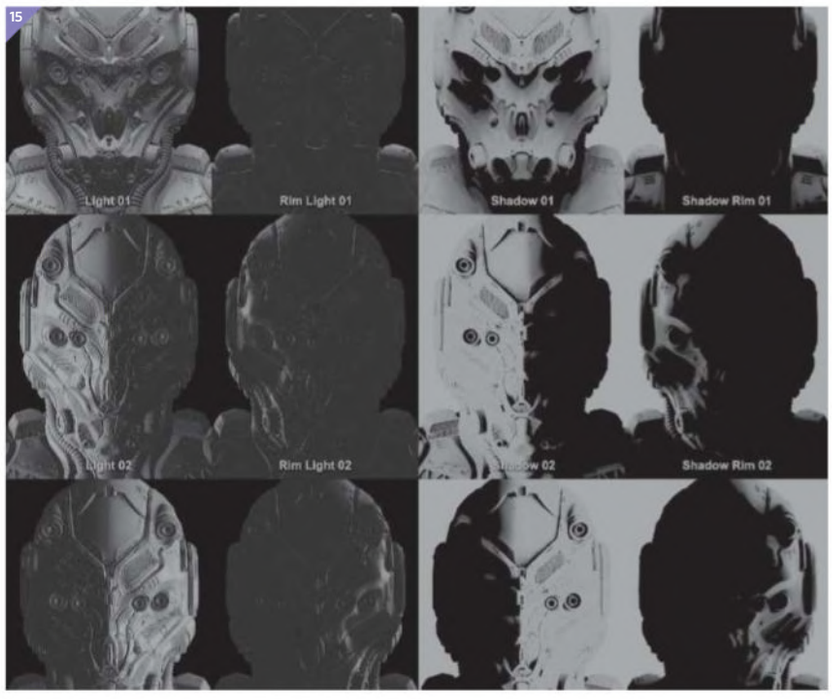
14 PBR render pass Start by finding a good camera angle in ZBrush. Use ZAppLink to store it, or simply create an animation key on the timeline. The goal here is to have enough diversity in the render pass to be able to generate the type of material you want in Photoshop. The more renders you have, the more you will be able to experiment different settings. The shadows will be rendered one time in the next step and composited directly in the final picture. Don't forget to disable them in your Render Parameters tab to optimise the time of your renders.

15 PBR lighting pass Create a new light in your Light tab, and push the intensity value to 3 to have a strong directional light. In your BPR Shadow subpalette, increase the level of rays and angles for softened shadows. Once again, find a good setting to have both nice shadows and a decent render time. Make three different lighting passes from the following; the top, the left and the right. Repeat the process to generate three rim lights from the previous angles. Each of those lighting passes will have their own shadow pass, which will allow you to have a tighter control in Photoshop.

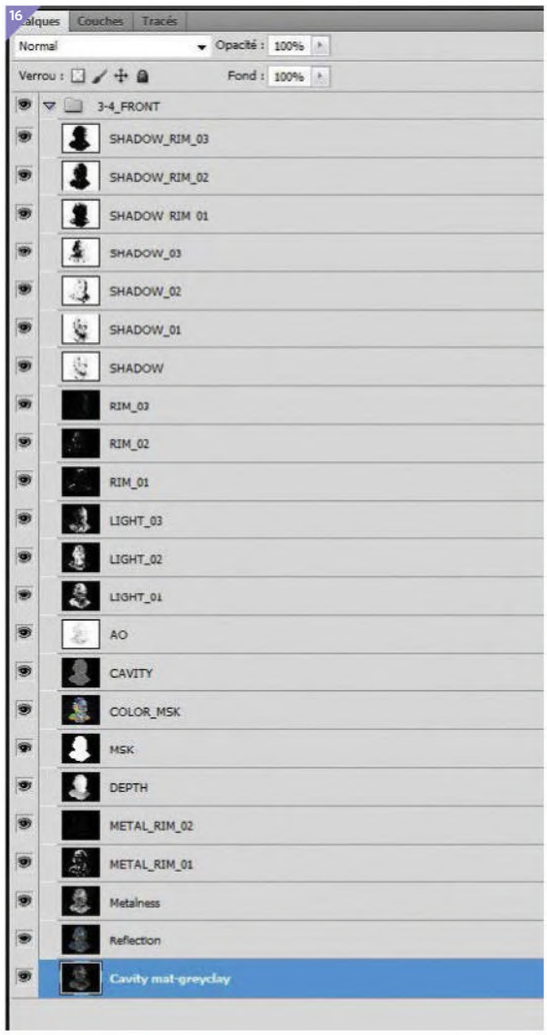


DynaMesh tricks

In ZBrush, the definition of your DynaMesh is based on the scale of your object. If you're already DynaMeshing your SubTools in 2,048 and you want to have more definition, you can use this work around: in the Deformation subpalette, use the size slider to upscale the chosen element. Note that we recommend to upscale by 100 and to downscale by -100 to keep the same size ratio. Once the object is DynaMeshed at the desired definition you can downscale it and it will return to its original position.



16



16 Compositing in Photoshop Import all your different render passes in Photoshop and stack them into one single image. Rename all your layers to have an organised PSD. Keep the materials together as base and all the lights on top of it. Tweak the blending modes, the opacity slider and add a modifier layer as Hue/Saturation, Level or Brightness/Contrast. To change the colours of your different elements, use a colour mask and the Magic Wand tool to fill in the selection in a new layer. Once again, adjust the Blending mode and the opacity to get a good result. For the micro details use the Lasso and fill the selection with the chosen colour.

17 Special effects To generate a smoke effect in the back, use a black-and-white mask from ZBrush and use a Gaussian blur with a big percentage for a foggy silhouette. Then create a cloud in Photoshop, tweak the levels and the opacity, and then merge the two layers down. You can repeat the same process to slightly add smoke in front of your helmet using a lower opacity. Use a depth-of-field pass to generate a realistic blur based on your camera focal. To add particle effects and lens flare, don't hesitate to use HD stock photos from website like CG Textures (cgtextures.com). As a final touch, you can create chromatic aberration by shifting your RGB channels in a different direction. Create a fusion mask on your merged layer and erase the centre of the image to just get a nice effect on the border.

17

How to create a colour mask

To have perfect control and to stay really efficient when using Photoshop, we've created a useful colour mask using the Nicks Tools plugin. It is a really handy open source ZBrush plugin that can be used in any type of project. Instead of manually filling each SubTool with a different colour, it allows you to do it in one click. You can also enter a value to multiply the grow or shrink selection of a mask. The large list of functionality is amazing and it will greatly help you in certain case. Check it out at: **artofnickmiller.com/resources.html**.

BLENDER

Blender

Kevin Hays
artstation.com/artist/khays

A 3D hobbyist, Kevin has been freelance modelling over the years and has used Blender since 2007

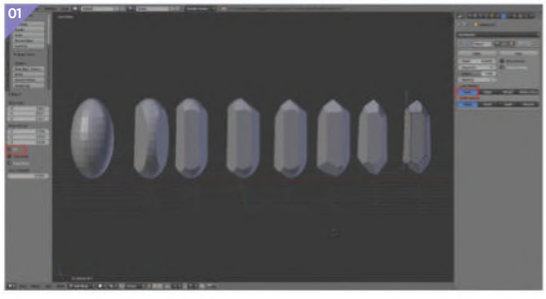
Model and texture crystals in Blender

When creating the crystal turtle project, we knew that the crystals on its back would be the centrepiece of the image, so creating a realistic glowing crystal was essential to the success of this project. In this tutorial we will guide you through the steps it took to creating these crystals.

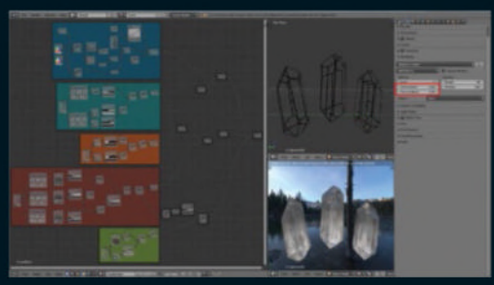
As with any 3D asset, the first step is modelling. We'll start by taking a look at how to use the Bisect tool to transform a sphere into a crystal model in just a few clicks. However, modelling won't be the main focus of this tutorial. What really makes these crystals convincing is how the material is set up, which will be the real meat of this tutorial.

We'll begin by covering the node setup and looking at the crystal's surface shader. By combining different coloured glass shaders, we can simulate how a prism splits light. Then we'll move on to volumetrics, which is split into three different steps. The first step is to give the crystal a vertical gradient, so that it's opaque towards the bottom and translucent towards the top. The next step is to add some speckled details using procedural textures. The last step is to use a 2D image texture and transform it into a 3D volumetric texture for even more detail on the insides of the crystals. Lastly we'll discuss how we did the surface displacement to add yet another level of realism.

01 Model the crystals We modelled the crystals using the Bisect tool, which cuts planar faces off the side of a model. By default, there is no shortcut hotkey for this tool, but you can create one or search for 'Bisect' in the spacebar search menu. Starting from a UV sphere, use BIsect to cut away faces from the object until there are no curved surfaces left. In Edit mode select everything, search for Bisect, then click and drag a line through your object. Anything outside of that line will be cut away. Make sure to check Fill in the tool panel so that the cut-away plane gets filled in. If you find that BIsect is removing parts you want to keep, drag the line in the opposite direction and it will remove the other side instead. If it doesn't seem to be working, apply rotation and scale in Object mode.



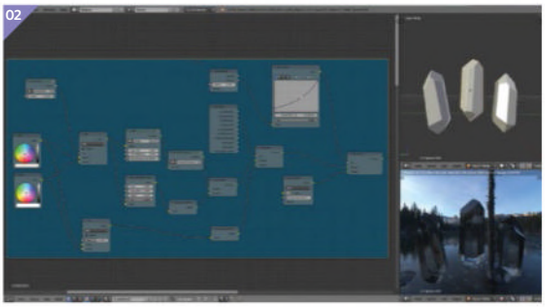
YOUR FREE DOWNLOADS
from filesilo.co.uk/bks-798
• Tutorial screenshots

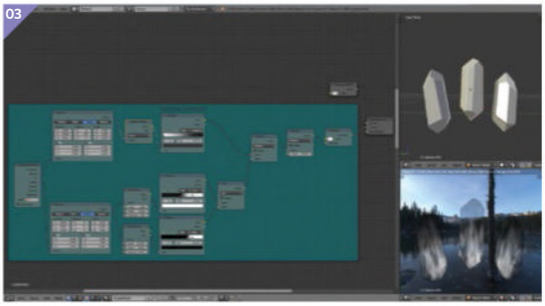


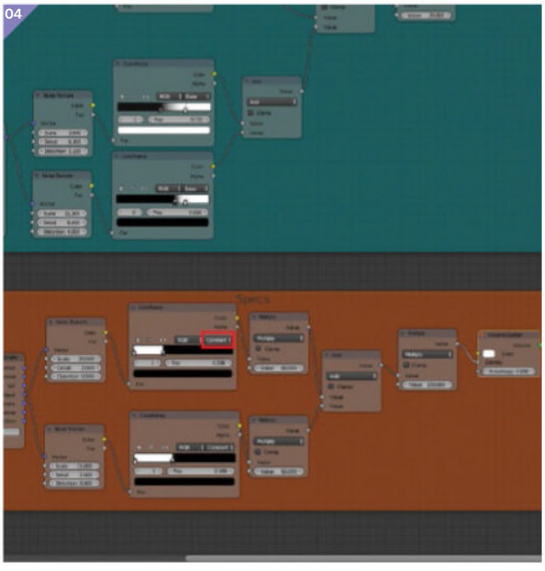
More glow and reducing noise

In our case, the glow we added with volumetrics wasn't enough. You can add more glow by adding a lamp inside the crystal but this can introduce a lot of noise to our renders. To reduce the noise without having to increase the number of samples, find the Sampling panel in the Render options, change Clamp Direct to a value of 4, and Clamp Indirect to a value of 1.

02

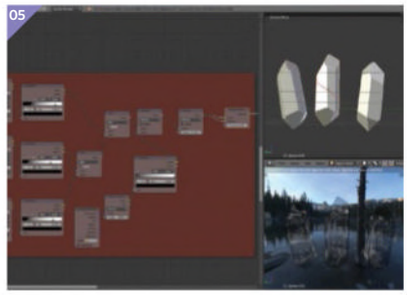






02 Surface material The key to this part of the shader is light dispersion. We want our crystals to split light like a prism does. To do this, add three glass shaders: one completely red, one completely green and one completely blue. Give each of these nodes a slightly different IOR value so that it bends light a little differently. Combine the different coloured glasses with two Add Shader nodes for a white glass that splits colours like a prism. You'll see that we've merged this set of nodes into a single group called 'Glass' in our node setup. You'll also see that our node setup replaces shadow rays with transparency. This isn't physically accurate, but we do this because we don't want the shell of the crystal casting a shadow on the inner volumetric details that we'll be adding later. Not only does this speed up render times and reduce noise, but it also helps make the details easier to see. Also note: whenever working on a shader that reflects or refracts light, it's always a good idea to use an environment HDRI.

03 Volumetric vertical gradient When looking at references, we noticed a lot of crystals were more opaque towards the bottom and more transparent towards the top. We can reproduce this look with volumetric textures. Using generated texture coordinates, add a Gradient Texture node. Then modify its direction and falloff using a ColorRamp node. You can hook this up to the Strength socket of an Emission node, and add that to the Volume socket of our material output. This looks a bit too uniform, so add some Noise Texture nodes and multiply their output with the gradient texture. When working on the volumetrics replace our Glass Shader node setup with a simple temporary transparent shader for faster render times.

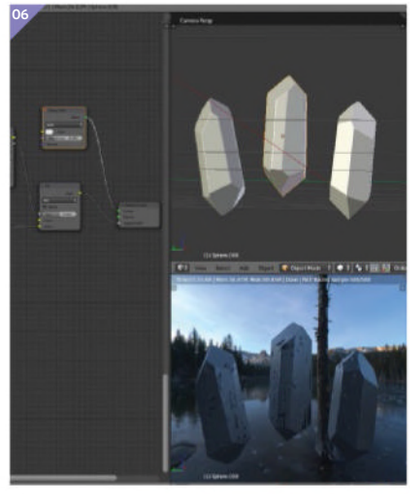


04 Volumetric specs We can add some irregularity and visual interest to our crystals with more noise texture nodes. Add a Noise Texture node and run it through a ColorRamp node with a Constant falloff for a hard edge. For more variety, we duplicated the Noise Texture and ColorRamp nodes and gave the second set a slightly different scale and ColorRamp falloff. After combining these two noise textures with the Add node, plug this output into the Density socket of a Volume Scatter node. This can be combined with our vertical gradient using the Add Shader.

> We do this because we don't want the shell of the crystal casting a shadow on the inner volumetric details that we'll be adding later

05 Volumetric image texture This is the final layer of detail for our volumetrics. Since there aren't many options for procedural textures in cycles, we tried using a grunge image texture. As our image is only 2D and we want to map it to a 3D volumetric space, we split up the texture coordinate into three different vector mapping nodes – each with a 90-degree rotation along a different axis. Then the transformed texture coordinate is plugged into our image texture, and combined back together with the Add node. We can multiply the output of our textures with a Voroni texture to break up uniformity from our input image. Plug the output of this into a Volume Scatter node and combine it with the other volumetrics using Add Shader.

06 Surface displacement The last step in making these crystals look realistic is to add some surface detail. Here we can just use the generated texture coordinates with a grunge image texture and a noise texture, then combine the two with a colour-mixing node. When experimenting, we found we could combine this Bump map with the volumetric image textures from before – giving us an even better-looking output. While working on the surface displacement; it's a good idea to use a simple glossy shader for the material surface. This should make it easier to see the surface details you're working on.

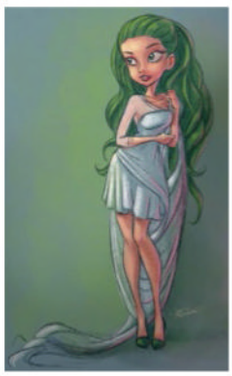


CARLOS ORTEGA ELIZALDE
Green Morning, 2015

Software
Maya, Mudbox, Photoshop

Learn how to
- Create an appealing cartoon character in basic steps
- Make a graceful silhouette
- Create a custom hairstyle
- Set up a lighting rig to showcase the model
- Finish the basic image in Photoshop using passes from mental ray

Concept
I've always loved the elegant mix of organic ornaments and fabrics that are frequent in the art nouveau style, this was in part inspiration for this piece, which started out originally as a quick sketch.



YOUR FREE DOWNLOADS
from **filesilo.co.uk/bks-798**
- Scene files including lighting setup
- Tutorial screenshots

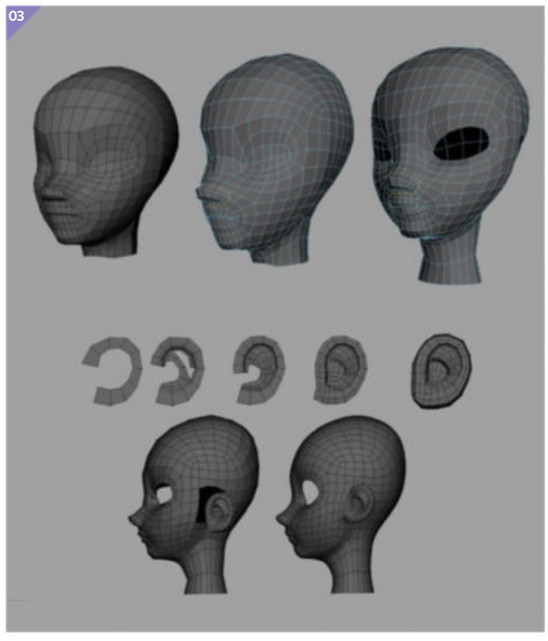
Create a convincing cartoon character

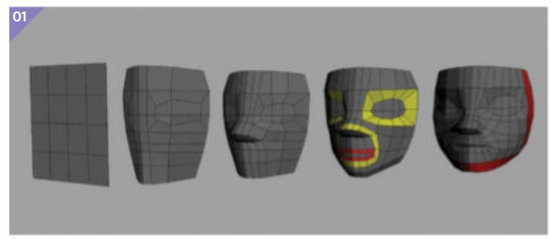
Render a charming cartoon pin-up, prepare a hairstyle and complement a scene with organic ornaments and lighting

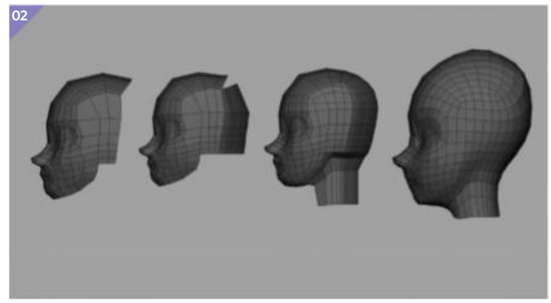
This guide covers a basic workflow used for creating a female cartoon pin-up. It covers areas such as a basic way to start a model and important ones such as creating hair. While there are tons of different methods, here we show you a quick and flexible way to define a hairstyle – without worrying too much about technical aspects so you can focus more on the shape, silhouette and other creative aspects. We will work with Maya, Mudbox and Photoshop, but most of the technical aspects and all of the artistic parts can be done in any other equivalent software.

01 Model the head Starting with a 4x4 plane we start by creating the edge loops that will become the main areas of the face, such as the eyes, mouth and nose. The eye sockets are just circular extrusions and the mouth consists of two circular extrusions: an outer one going around the mouth up to the nose bridge area, and an inner extrusion that will become the mouth and lips. At this point it is useful to have a sphere to serve as an eye placeholder – it will also help us to build and define the eye sockets around it.

02 Close the head By extruding the border edges of the sides and the bottom of the head we will get the polygon flow that will define the jawline. Now extrude the edges of the sides and the top to the back of the head, this will leave room at the bottom to extrude the neck. You can close the head by extruding the edges of the back and merging the vertices to the top of the head – just think of the shape as a smoothed cube. Then extrude the hole in the bottom of the head to create the neck. Using the Sculpt tool, smooth the new edges to get a round shape.



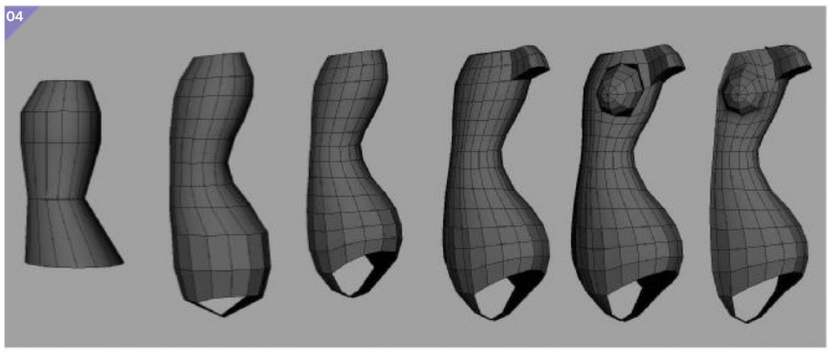


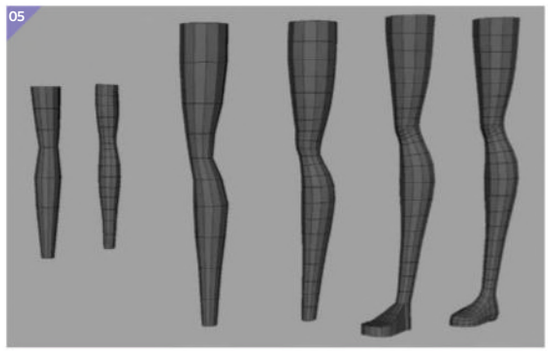


03 Model the nose and ear We create the nose by adding a face loop in the nose area. The ear is modelled in parts, starting with a simple row of polygons shaped as a spiral section. We don't need a fully detailed ear for this character – modelling a simple, suggestive antihelix and the inner ear is enough to get believable cartoon ear. By deleting some faces on the head and leaving the same amount of edges in the hole and in the ear border, we can attach it to the head using a Bridge operation.

Take control of every vertex created
A common error is trying to get a lot of detail from the start, if you get the shape of the head you want with a small amount of geometry and a smart topology it becomes easier to predict when and where to add or delete new edge loops. That being said, don't be afraid to add new edge loops for further detailing without the fear of losing the shape you already had. Tools like Edit Edge Flow, the Slide Edge Tool and the relax and smooth operations in the Sculpt Geometry tool are really powerful for smoothing out an organic, dense mesh.

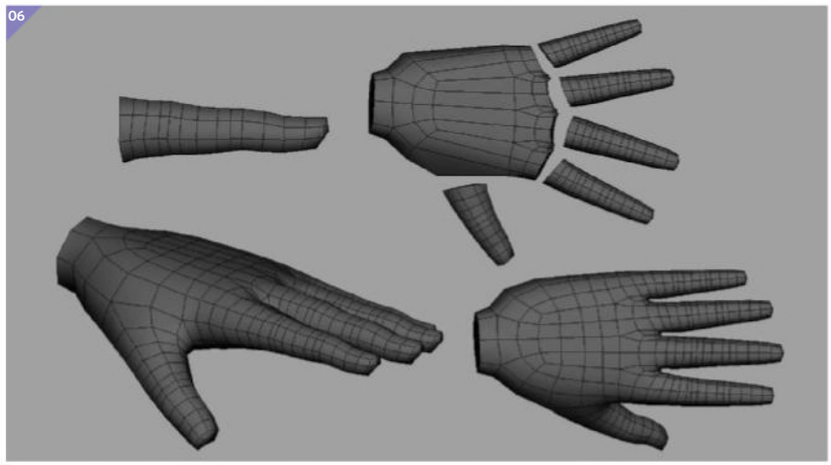
> If you get the shape of the head you want with a small amount of geometry and a smart topology it becomes easier to predict when and where to add or delete new edge loops



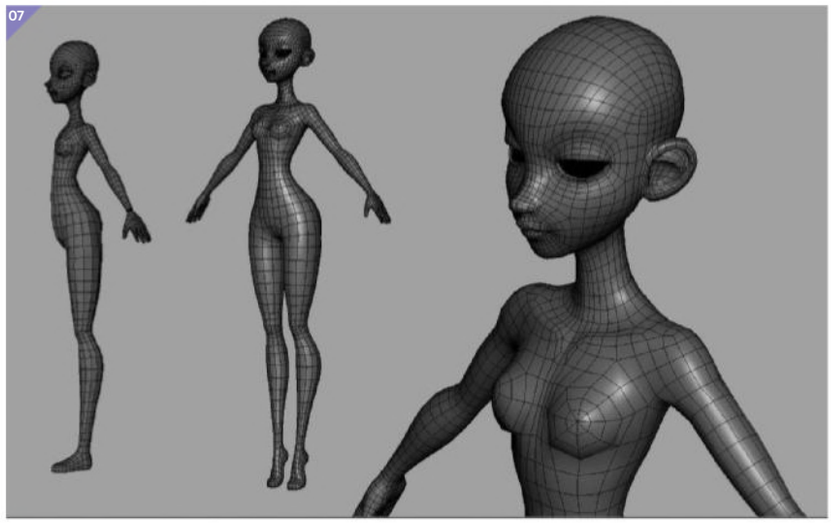


04 Build the torso The main shape of the torso is basically a cylinder. With a few divisions we start to block the base of the neck, the shoulders, the waist and the hips. With those areas defined it's easy to start adding divisions to the rest of the piece, as well as extruding the shoulders. The breast area is just a section of a 8x8 sphere attached in the front, slightly below the shoulder's height. We close the bottom area by creating a bridge with the central edges in the front and back, this will leave the holes for attaching the legs later.

05 Build the arms, legs and feet The arms and legs are cylinders too, but with the edges extruded. The main areas and widths that we want to define here for the arm are the biceps, elbow, forearm and wrist; and for the leg the width of the thigh, knee, calf and ankle. For both parts always check from both views to get the desired shape. Once we have those sections blocked we can start detailing by adding more divisions. The foot in this case is created by closing the bottom of the leg and extruding the front faces, we don't need toes since our character will have shoes.

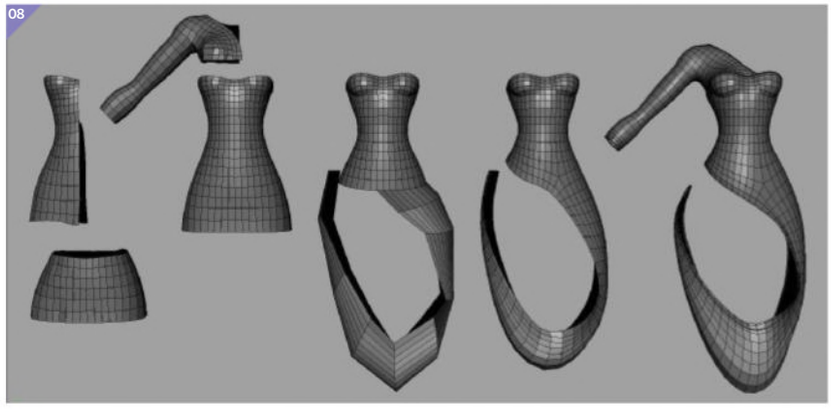


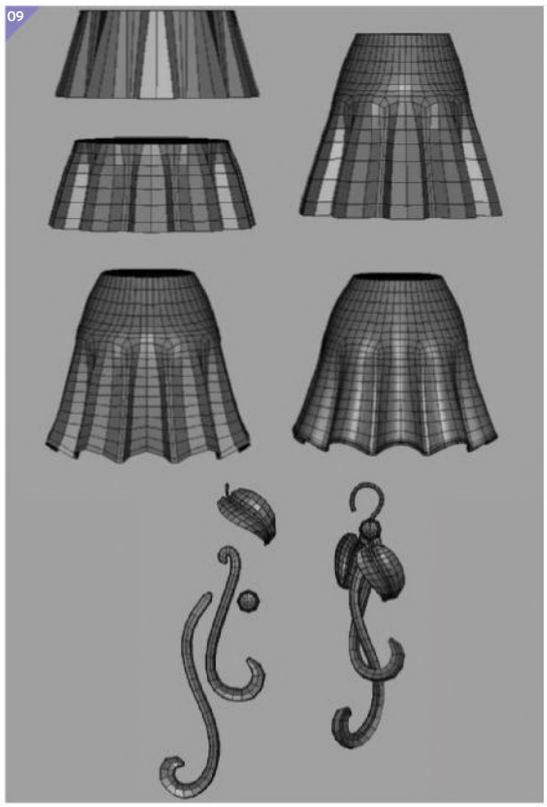
06 Model the hands Starting with a cylinder we create a base finger. The palm can be started from a cube with enough divisions for attaching fingers later. It can come in handy too, when adding the skin, to leave space between each finger. Duplicate the finger and place them around the palm, scaling each one to its proper size. Then attach each finger to the palm, using a Bridge or the Merge Vertex Tool. The base topology is really basic, so focus more on avoiding a blocky shape by sculpting or tweaking vertices using a soft selection.



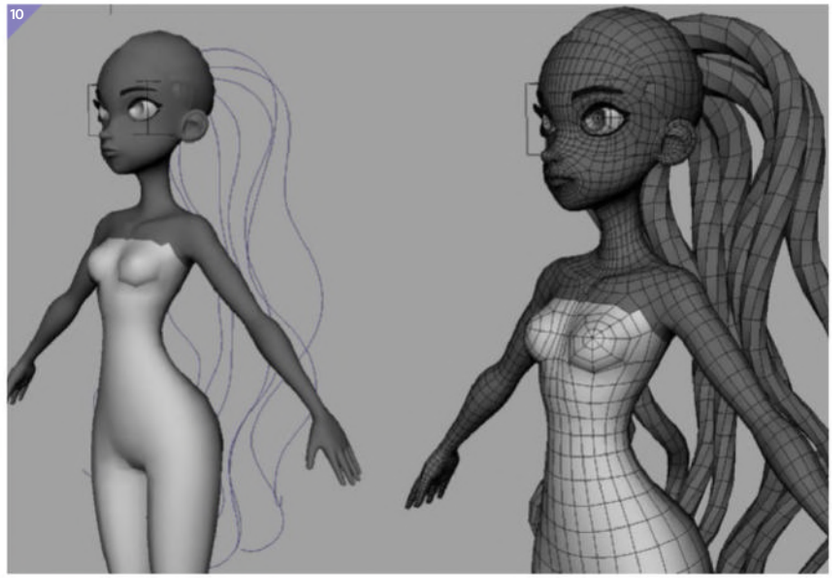
07 Merge the pieces together Attach all the limbs to the torso, and smooth out the joins if needed. Add some clavicle detail on the body since it is a part of the body that will not be fully covered by cloth. For the final eyes we start with a sphere, extruding and sculpting the bulge on the front that is useful to catch highlights pleasantly. The iris and pupil is just a concave disc placed inside the outer eye, by using a transparency map it will enable us to see through the cornea inside the iris.

08 Make the dress We create the dress by using a duplicate of the body, this way we will make sure it will fit with no major effort. The upper part of the dress uses the geometry of the waist up to the shoulders, and the skirt itself will come out of the hips' geometry. Once extracted, we need to get rid of the breast details and close the dress more naturally. The cloth extension is just an extrusion of the bottom half of the dress, going back and up to connect to the opposite side of the dress at the hip's height.





09 Add the skirt and accessories The skirt is created in two parts, the folds can be easily created by using a cylinder and pulling out vertices to create the folds. After extruding and shaping the bottom part of the skirt attach it to the hip's geometry that was previously extracted from the body. The shoes are created using geometry from the feet, and then building them around the foot. For things like earrings and hair ornaments, model the elements and then combine them to create each accessory, this adds a consistent style by having repeating patterns across the model.



Plan the hairstyle

You can easily block the hair geometry using polygons by thinking of hair as broad shapes or clumps instead of hundreds of individual elements. By keeping a 'grid' or 'patch' topology on the geometry of the strands, it can lead to different styles of output depending on your needs. You can use the geometry to get an appealing cartoon look, use transparency and bump maps to get a more detailed look, or create curves from the geometry to grow a more complex and dynamic hair system.

10 Build the hairstyle The hair here is divided into two parts – one covers the skull and the other is formed of long strands in the back. For the calvaria we can start with a sphere and delete the poles to easily get the basic strand shapes coming from the forehead to the back of the head. Keeping the topology as grid patches is useful for converting later to a Maya hair system. For the long strands of hair in the back draw the main shapes using CV curves. Then, build blocks of hair by extruding a circular plane along the curves.

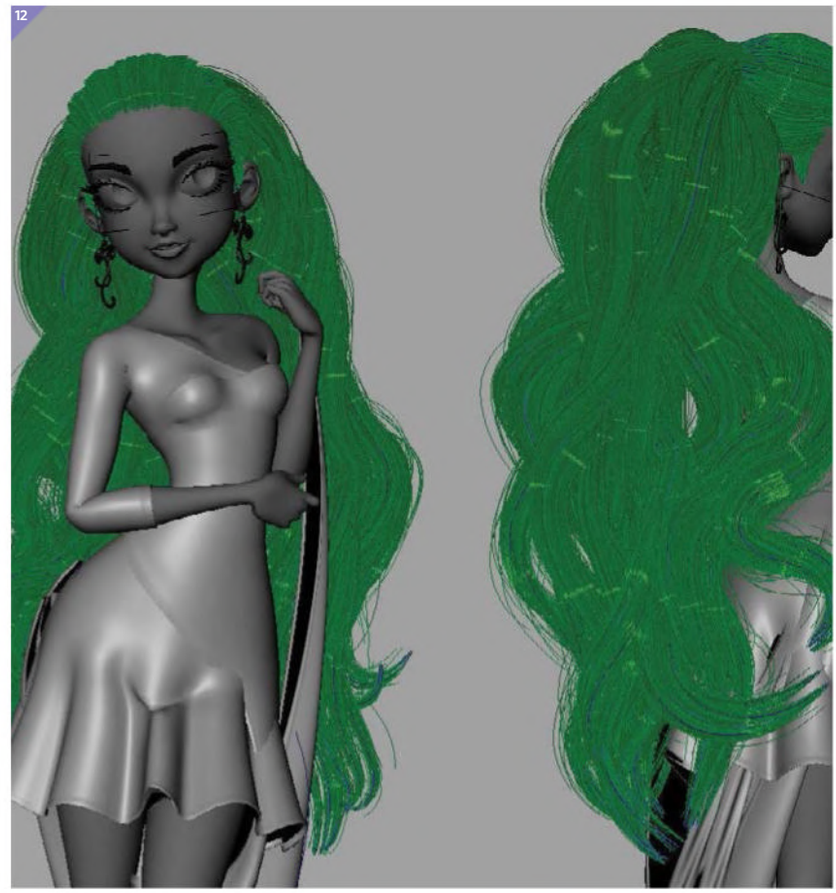
11 Detail the hairstyle Use the taper and twist functions in the Extrude options and sculpt the resulting mesh to get variations in volume and shape. Now extract the curves that will serve as guides for the hair. Select the Edge Loop and go to Modify>Convert>Polygon Edges To Curve. You will need to do this for all the necessary curves. There are free scripts on Creative Crash that can automate this process for you. Too many curves can lead to a heavy system – conversely, too few curves can force you to grow wider clumps with lots of hair to fill the gaps, and this will cause you to lose the silhouette of your hairstyle.

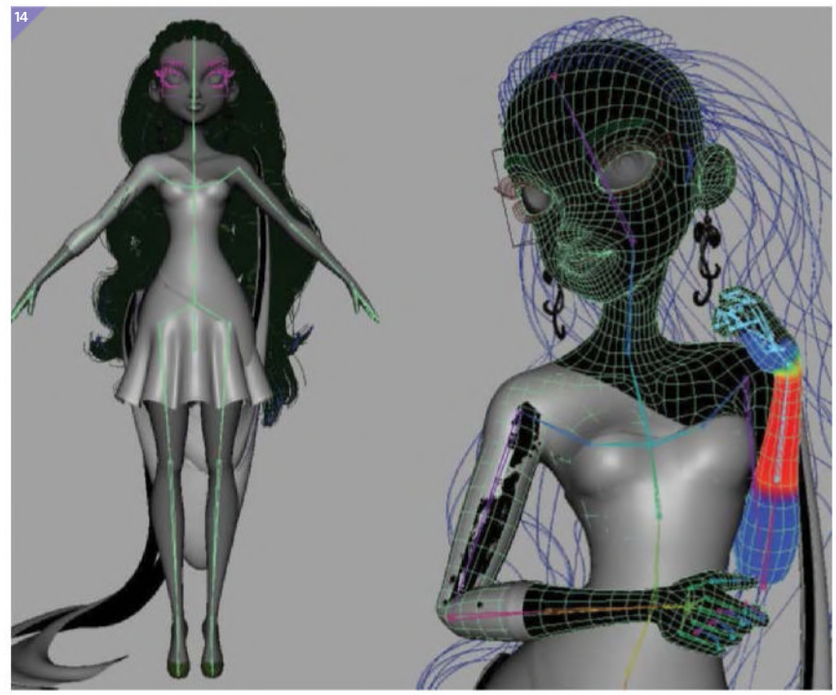
12 Finish the hairstyle Next step is creating two hair systems, one for the head and another for the long strands. Select the head curves and go to nDynamics>Hair>Assign Hair System, then assign a PaintFX brush to Hair. Now you can adjust the clumps' width and shape, hair thinning, number and width of hairs per clump, and hair shading. Keeping the original mesh of the hair without deleting history in the curves lets you modify the polymesh directly to tweak the hair and transfer those changes to the hair system. By default the hair gets dynamic properties, and you can change it to static for a still image like this.

13 UVs and texturing The improved unfold tools in Maya will let you unfold the body and cloth meshes easily, one UV tile per material will work great for this model. Moving to Mudbox, paint the textures of the character. You can start with a skin colour base and add yellow, orange and pink tones in the nose and ear areas, and purple tones in the eyes. Using references is a good way to know where those tones suit better. Now paint more maps such as a deep Scatter, Specular and Glossiness map. We will plug those to our skin material.

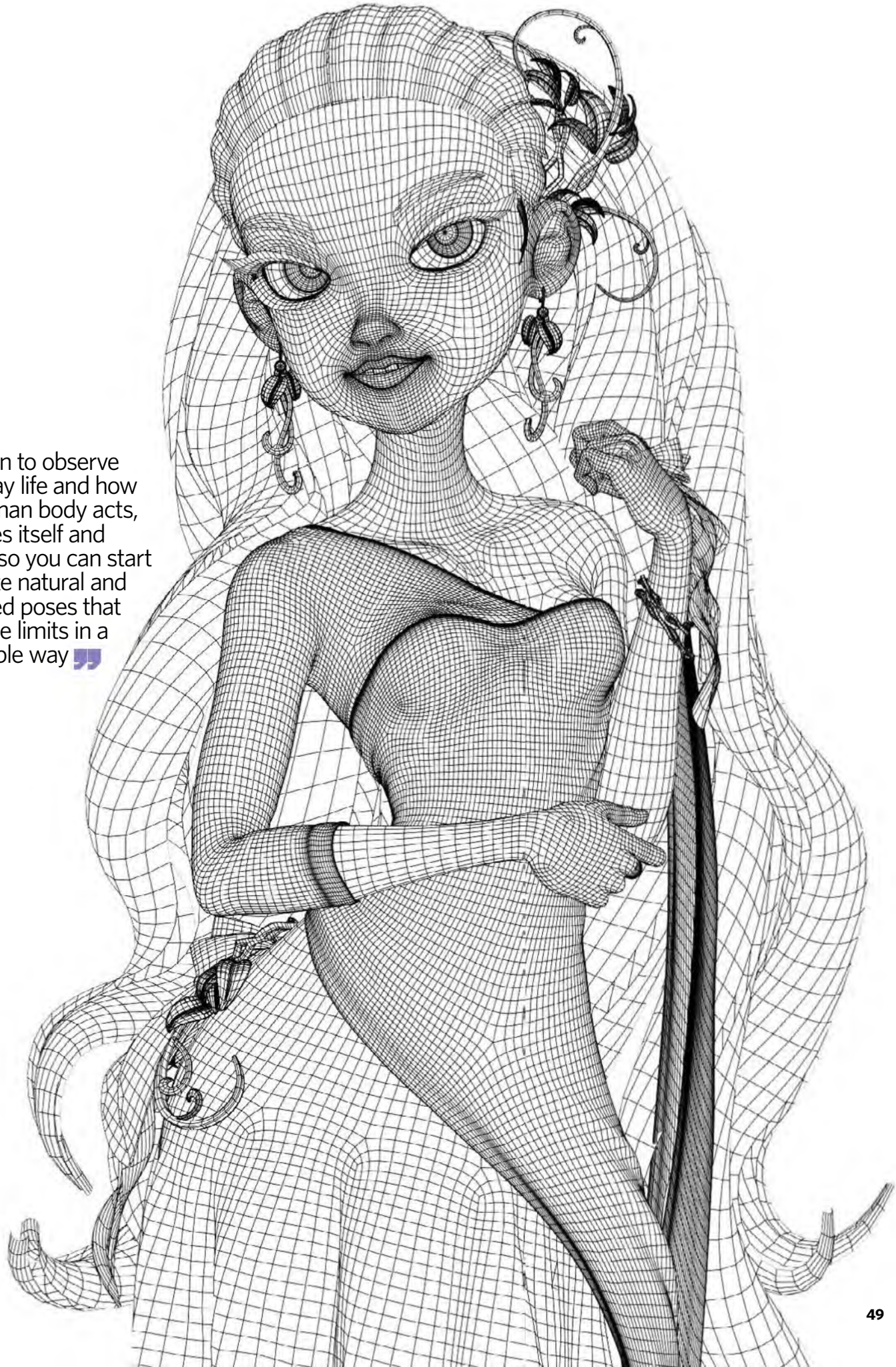
Getting a graceful silhouette

Learn to observe everyday life and how the human body acts, balances itself and moves so you can start to create natural and balanced poses that pushes limits in a believable way. Here, the shoulders tilt in the opposite direction to the hips to compensate the weight. The pose is almost an S shape, something playful and flirty yet balanced. Along with the flow of the shapes of the dress, it can help you to get a solid composition and guide the viewer from the face, down to the fabric on the ground.

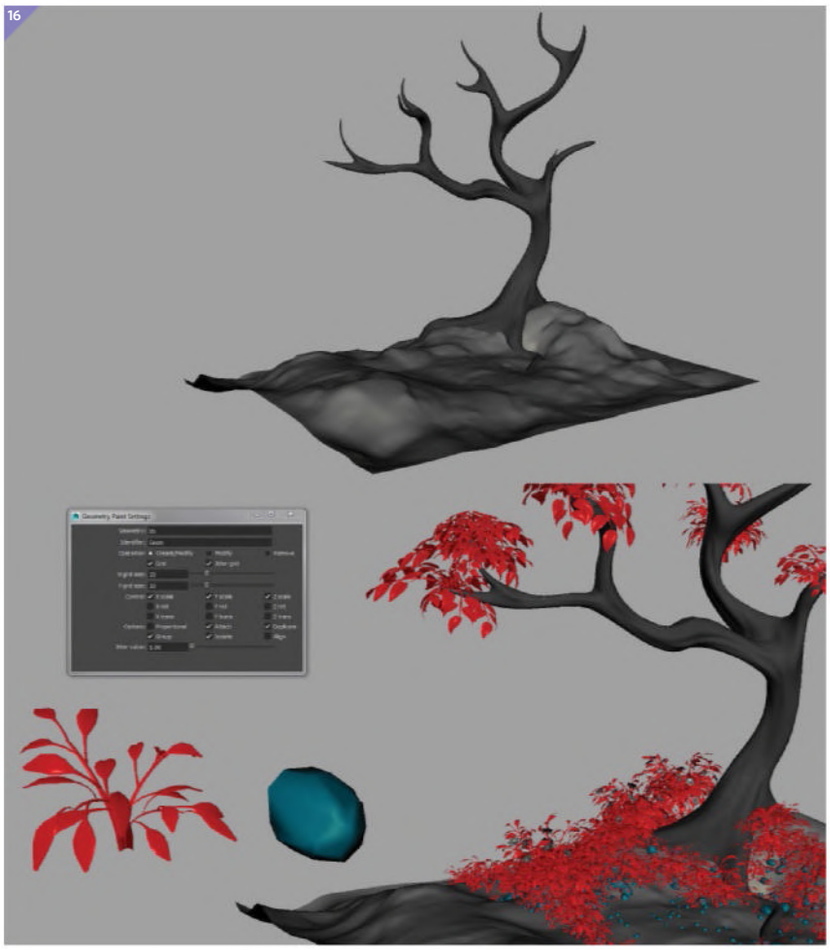




14 Pose the model Create a basic skeleton in Maya, drawing the joints from the hips up to the tip of the head, then legs and arms. Using the Snap to Projected Center feature will enable you to easily create the joints in tricky areas such as inside the arms and fingers. We bind the skeleton to the main mesh using the Heat Map option and Quaternion method, and since it is a skinny character this option works well by default. We can then skin the clothes and copy the influence of the body to the dress.



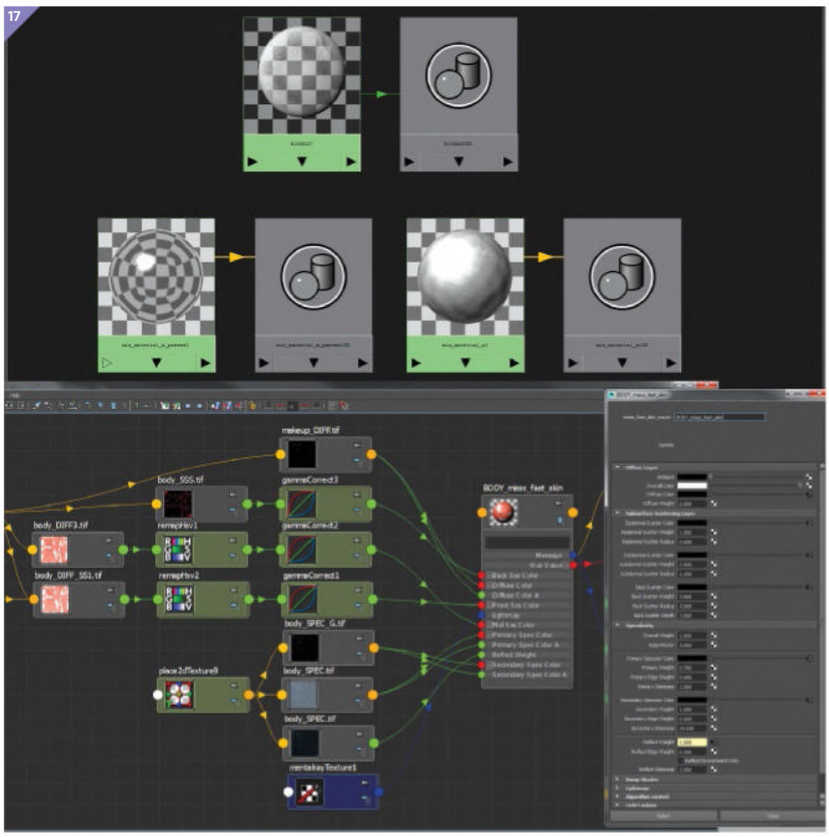
> Learn to observe everyday life and how the human body acts, balances itself and moves so you can start to create natural and balanced poses that push the limits in a believable way



15 Detail the model Once posed, send the character and dress to Mudbox for detailing to fix the weighting and volume loss for a more natural bending in the elbows, fingers, neck and knees. We subdivide the dress a couple of times, sculpt the foldings in the fabric and fix interpenetrations between elements. Most of the silhouettes of the objects were solved with pure polygon modelling, so at this point, bake some normal maps of the cloth to get all the folding details. Then export all the elements with one level of subdivision to save some render time and avoid the use of displacements.

16 Lay out the scene To complement the scene we will add some natural elements that will suit the atmosphere and colour palette. We will add a simple terrain which we can sculpt and detail in Mudbox. The tree can be quickly created in ZBrush – we don't need to overdetail the model, since the whole feel of the scene is smooth and more like a fake set rather than a realistic scenario. To fill the ground with details we use the Paint Geometry Tool which you can find in Maya Bonus Tools 2015. Its ease of use will let you quickly grow some plants and throw some rocks on the ground.

17 Materials in mental ray This scene uses the mia-material-X, the misss-fast_skin shader and blinn materials. It is useful to start playing with the Scale Conversion option on the skin shader to make it work, since the shader will depend of the size of your object. Plug the textures and be sure to gamma-correct each texture or work your scene with the Color Management option. The fabric material uses some ramps to define transparency and colour, so that it becomes more transparent and is at a glancing angle. Use glossy refractions objects, which will become blurred when seen through the fabric.

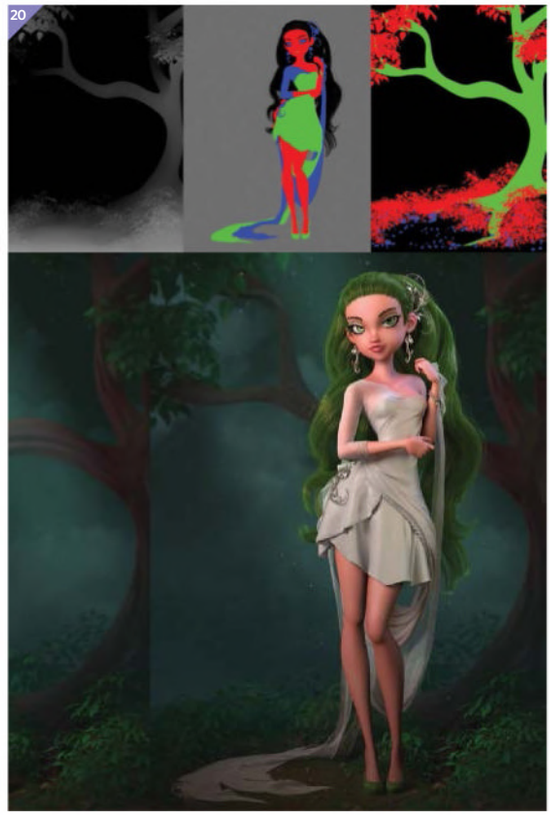


18 Lighting The light rig consists of four lights: key light, (spotlight) which gets most of the volumes and defines the main source of light; backlight (spotlight) to get some highlights on the hair on the left side; bounce light (arealight) to fill black areas; and environment light (IBL), which is set to a very low value – it fills the whole scene and casts reflections on the objects with a free blurred HDRI.

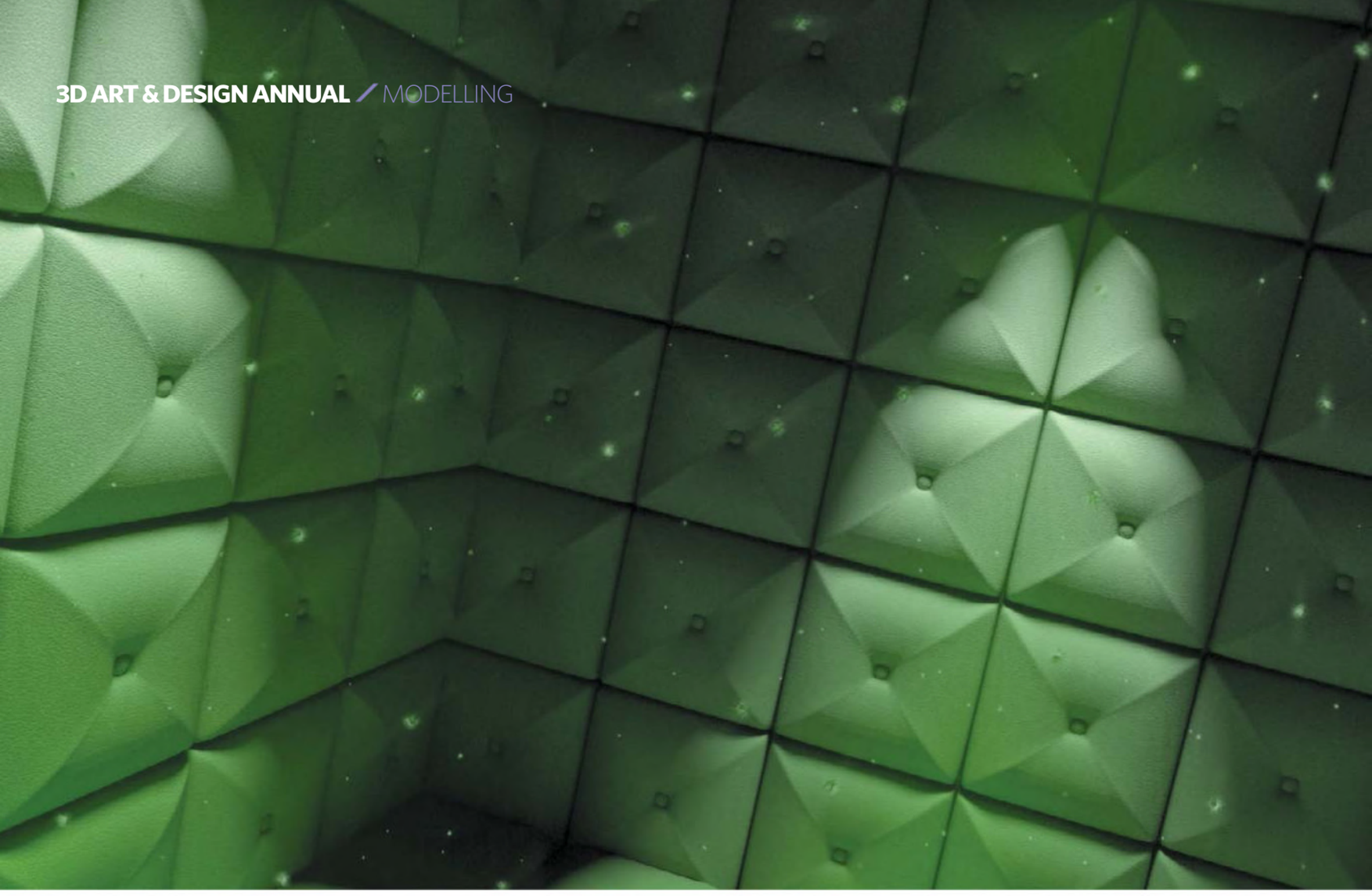
Be curious with the attributes

It can take a while to get a decent look with materials like the skin shader. A suggestion for this kind of cartoon character is to not look for realistic results or fixed recipes of values. Be curious, take your time to do various render tests with default settings and analyse every value by turning it off or changing its value to see the effect on the material. Try to be a good observer of real-world materials so you can replicate them in the future.

19 Rendering The scene is rendered with mental ray and Final Gather. For large resolution renders like this, it's useful to bake a Final Gather map at a lower resolution and then freeze it to save some render time. By default the skin shader takes into account the resolution of your scene to compute its light maps, be aware of this when doing large tests or final renders. For the same reason, we will render the scenario and the character in separate render layers using a 32-bit EXR format.



20 Finish the image Some useful layers to render during image compositing are a good set of masks, a depth pass and in some cases a normal pass. In Photoshop, use the depth pass, and add atmospheric elements such as fog and depth of field (using the Lens Blur filter). The masks are useful for adjusting the colour or exposure of different elements without struggling with selections. We can create light effects by painting with soft brushes and setting layers to a Screen or Add mode. The dust particles are just brushes with high scatter values and different blur settings for a feeling of depth.



Scatter a scene in Houdini

Houdini

Rainer Duda
rainer-d.de

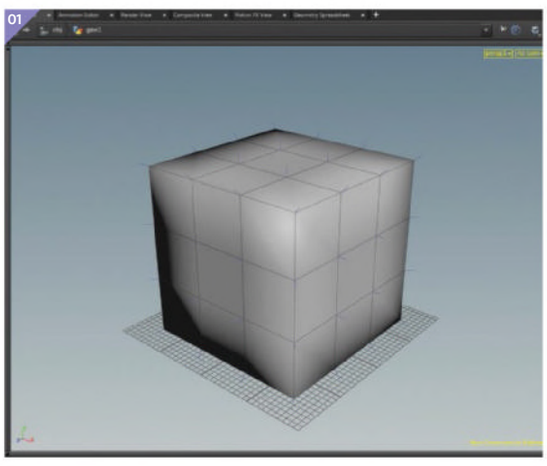
Rainer Duda is a Houdini TD who specialises in advanced environment creation techniques and enjoys life to the full

In this tutorial we take a closer look at Houdini's capabilities when it comes to scene construction techniques. We will create a small system in detail, which enables artists to create rooms without worrying about manually positioned assets. The keyword in our process is scattering, which is a procedure where an algorithm places objects to points spreaded around a specific input geometry. The core idea of scattering is fine but due to a lack of control regarding explicit positioning we will make use of a point array as scatter input in order to place tiles and deco objects. Therefore we will define a whiteboxing system which fits with all assets in the specific unit size of the scene.

We define a room by the use of a polygon cube geometry. After a subdivison step, the preparation starts and it will be time to cut the already visible tiles to separate primitives. Through an exploded view we have a better look at all the tiles which will make it easier to see the point numbers that we need later. Each separate tile will be subdivided again into four quads because we need the centre point in the bigger tile. This will be the anchor where the objects – the pillows – will be placed later. In addition to the pillows, which form the rubber cell, we place some lamps too.

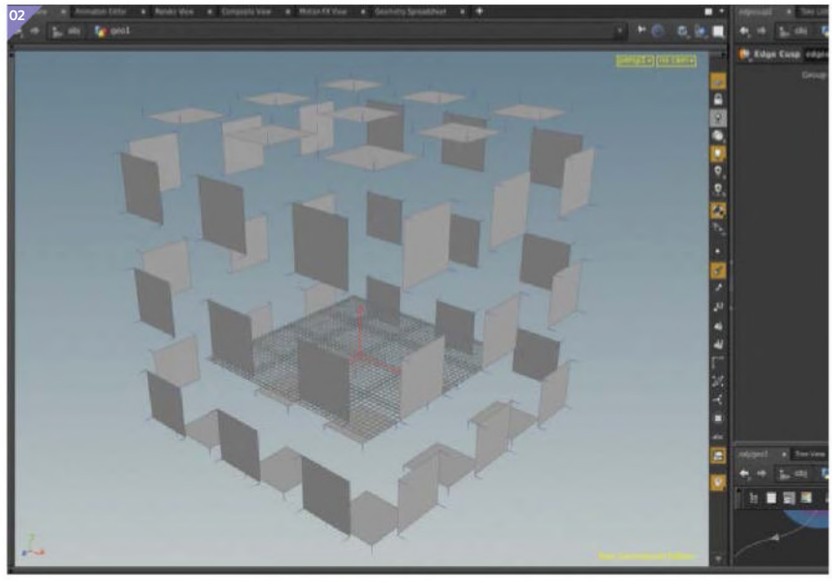
To tell the system where it needs to place all objects exactly we use a special method. We resort to the group SOP: an operator which lets us define a specific set of points which we can pack into a group. Afterwards it's time to feed two copy SOPs – the magic placement. The copy SOP takes two inputs. First, an object will be placed and secondly there will be a template for where it will be placed. That's basically a primitive input, but by adding the group name which we defined earlier its easy to set the correct positions.

01 Build the cell We start off by creating a geometry node and delete the node inside. Instead we need a box node. For primitive type we choose polygon mesh for an initial subdivision. Now set the size for x, y and z in each case to 3. To align the box with the grid we use a simple expression. In the centre field for the y axis, so just type: 'ch("sizey")/2', which basically means that we set the centre to a value that is half of the box's height. This is a solution which lets you easily position your objects on the ground plane.



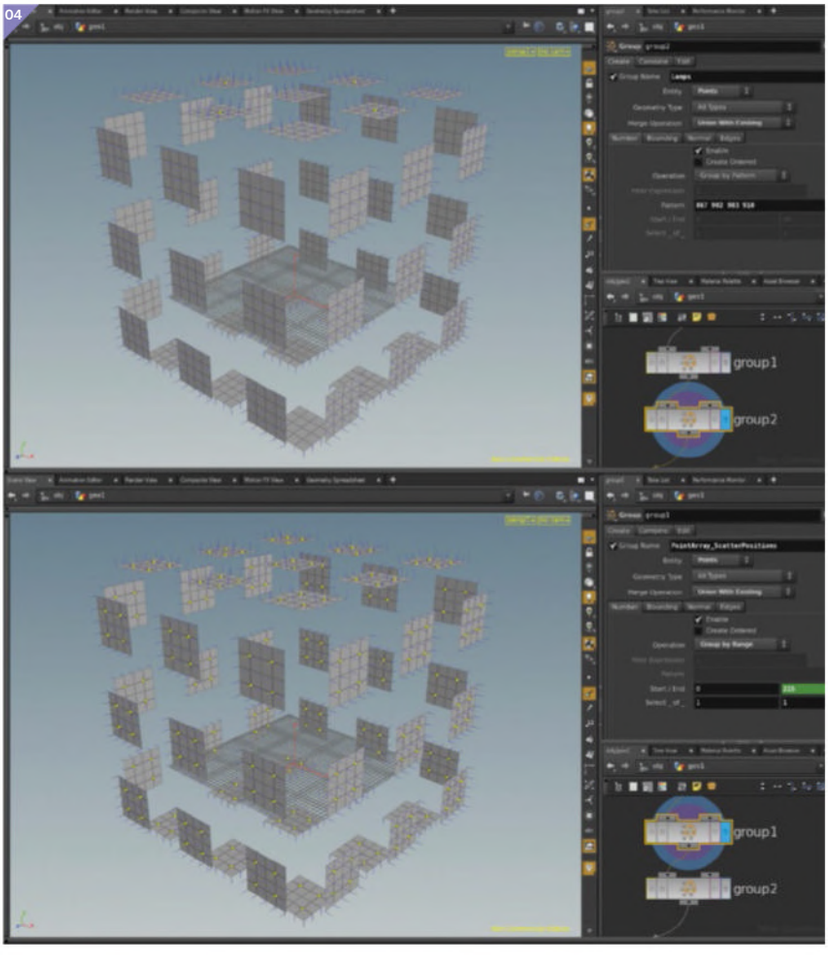
YOUR FREE DOWNLOADS
from **filesilo.co.uk/bks-798**
- Scene files
- Video tutorial

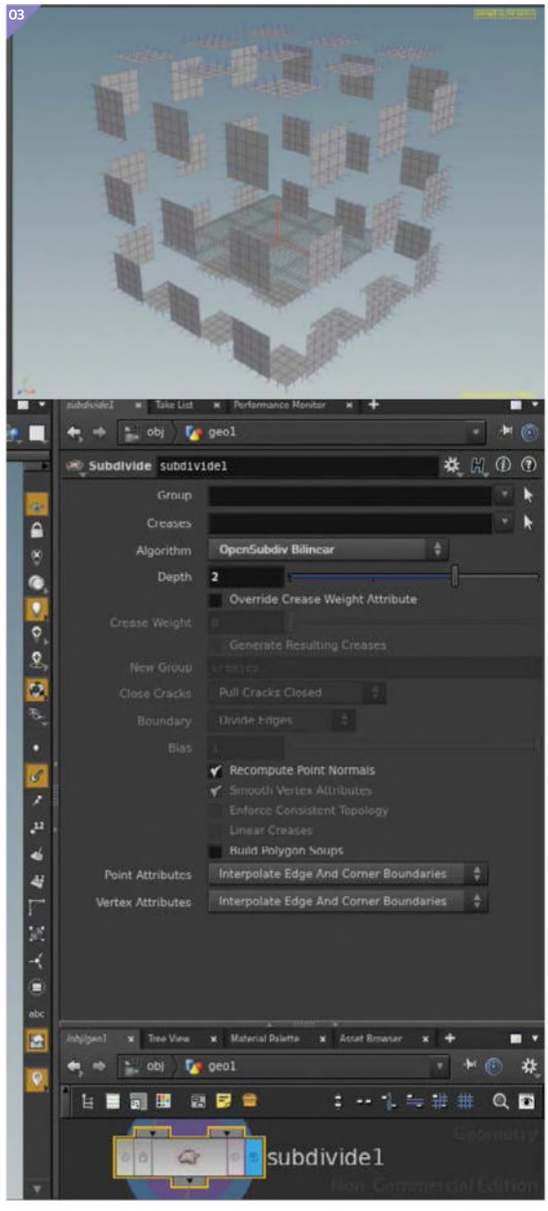
02 Intelligent slicing of the tiles Our next step is the creation of an Edge Cusp node. This SOP takes an input geometry, in this case our box, and splits the faces. Usually this is used to fake hard edges on an object but technically it splits the faces to separate objects. This principle will be easier to understand in our next step. Let's add an exploded view node to the Edge Cusp node. Both nodes don't really need additional changes. Just create and append them one after another to see all single tiles at the end. These are not subdivided here – they are single planes.



03 Subdivison and tiles Now it's time to prepare each tile with a specific amount of subdivisons because we want to give the copy SOPs a clear base to scatter the pillows. So, we will need to append an subdivide SOP node. The result by now is far away from what we expect. That's the result of the used algorithm. To get a proper result we change the algorithm to OpenSubdiv bilinear with a depth value of 2. Each tile should contain 16 quads. Now we are ready to start grouping points that we need.

04 Creation of logical point groups In this step we define explicit positions for the scattering. At first activate the Display Point Numbers function at the right side of the 3D viewport. Afterwards we create and append two Group nodes – one after another. The first Group node gets the Group name: 'PointArray_ScatterPositions'. The second Group node will collect points for the lamps. The Group name here will be 'Lamps'. Both nodes need a Merge Operation of 'Union With Existing'. In the first Group node activate the 'Group by Range' operation and set it to 0 – 215 with a step count 1 of 1. For the Lamps use 'Group By Pattern' and set the values to '867 902 903 910' in Pattern.

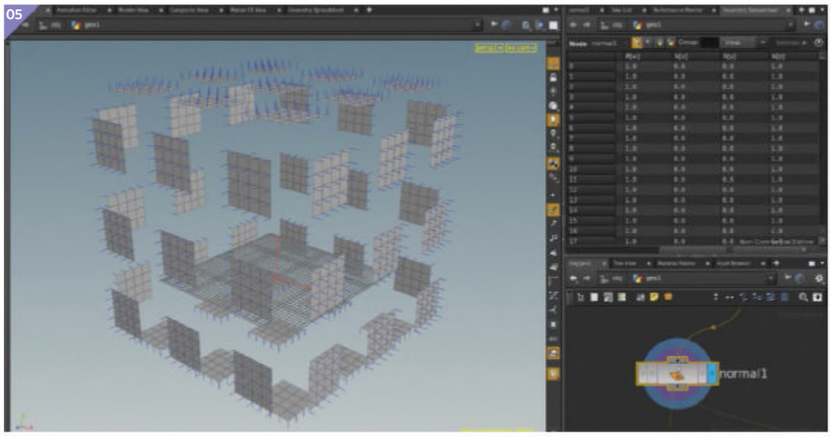


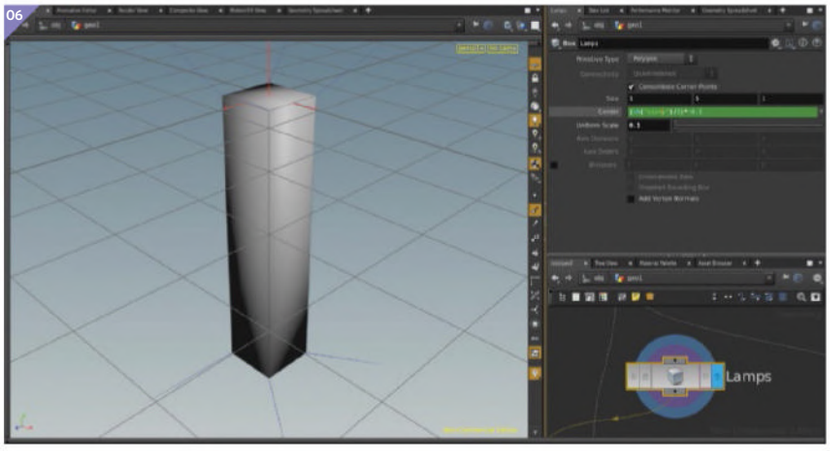


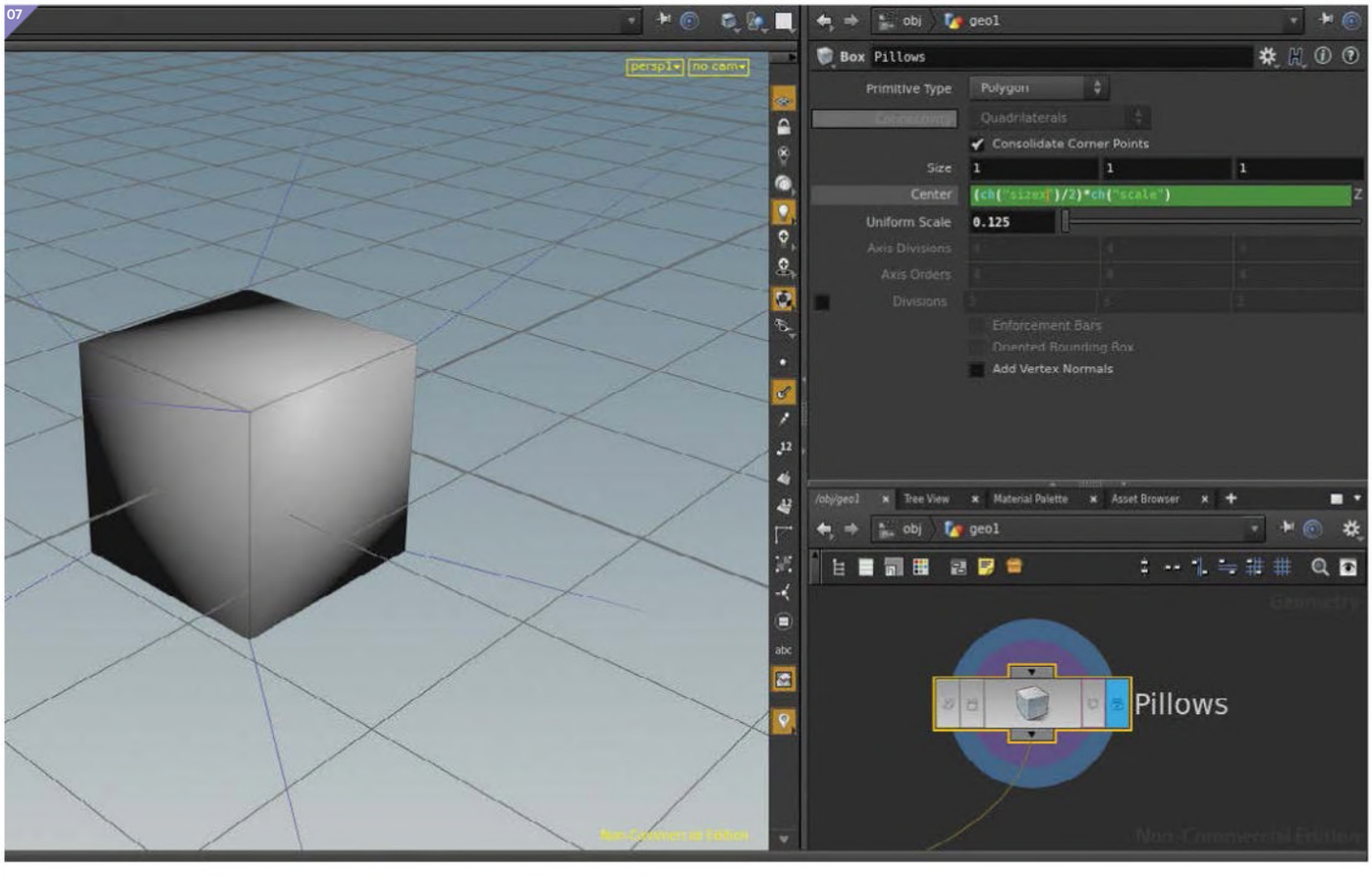
05 Add normal info to points What we have left on our scattering base now is the point normal information. To add them we simply create a normal SOP node and append it to the last Group node. With the default values we now have added valid normal information. Just add the normal information to points – this will be handy when we prepare the whiteboxing assets and their placement. This is also the number one choice when it comes to making an object with hard edges or soft edges without cutting the geometry at all.

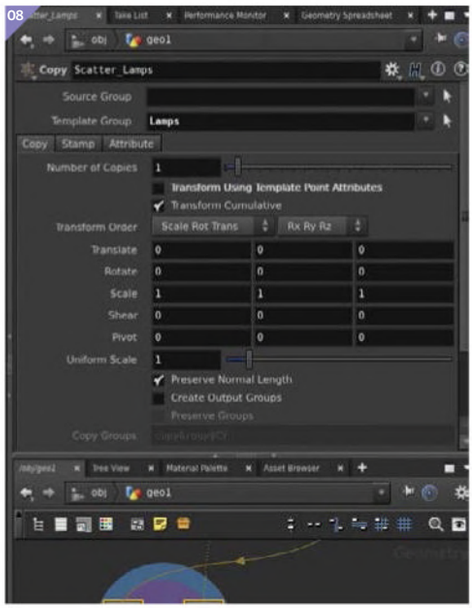
06 Whitebox to test the system Before we use final assets we make use of simple whiteboxing assets. Let's add two polygon boxes, name one 'lamps' and the other one 'pillows'. The lamps need a height of 5 but this time use the Uniform Scale parameter and shrink it down to 0.1. Last but not least, let's position the lamp on the grid via a centre of the Y expressions of '(ch("sizey")/2)*-0.1'. It is important here that we not only divide the overall height by two but also take care of the Uniform Scale which we had decreased. We need the minus value so that the lamps are facing down.

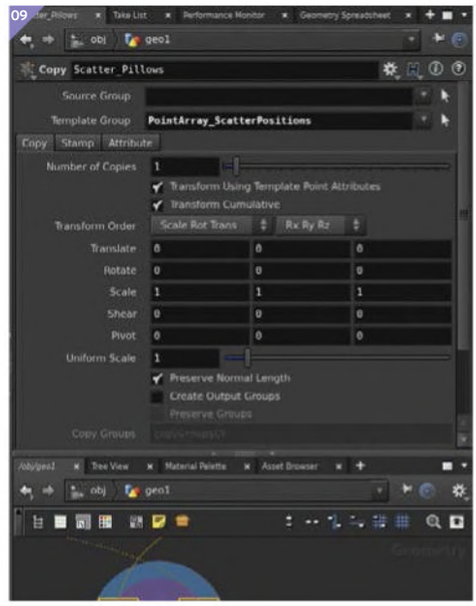
07 Whitebox the pillows The pillows have a quite similar setup to the lamps. We leave the overall size at one and just play a bit with the Uniform Scale parameter. We decrease it to 0.125, but this time we make something special. Instead of aligning the box to the ground plane we move it to the left from the viewer sight of the viewport. We go into the centre of the Z area and add the following expression: '(ch("sizex")/2)*ch("scale")'. Instead of typing the accurate value into an expression you can also use a relative reference of the copied parameter as well.









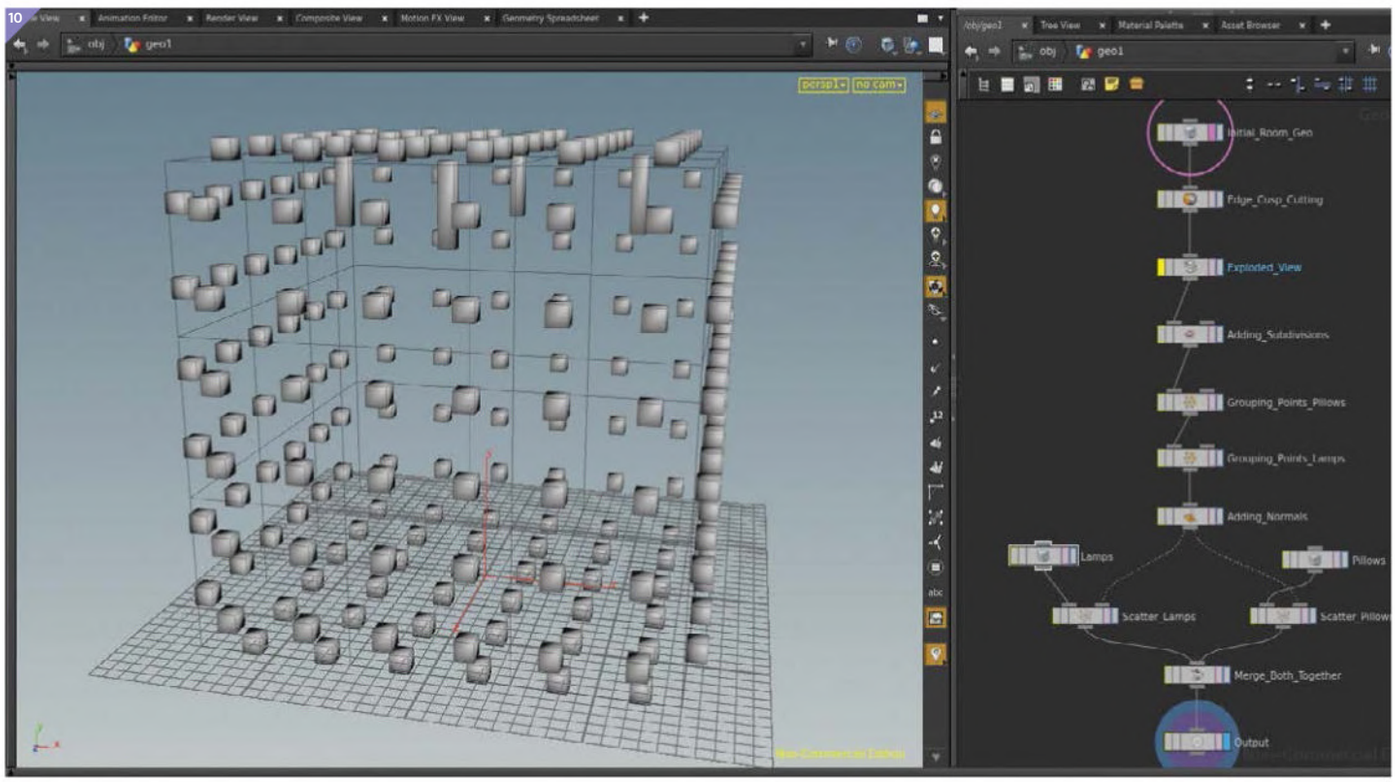


The scatter orientation by making use of attributes

We scattered objects along a self-made point array but the appearance is always the same for each tile. You can break through this if you want to rotate the tiles randomly in 90-degree angles. The copy SOP is one of the most powerful tools in Houdini and enables you to stamp various attributes across the scattered objects. You can randomly change material as well as texture attributes or the rotations to give your scene a more natural look when you work with colours and textures.

08 Bring in the copy SOPs Now what we have left is to spread the whiteboxed assets around are the copy SOPs. We need to create two of them. The first one will take care of the positioning of the lamps. Let's plug the lamps to the first input and the normal SOP to the second input. But before we move one, remove the hook at Transform using Template Point Attributes and add the name 'Lamps' into the Template Group slot. It will say that the copy SOP should only use the points in this group for the scattering.

09 Groundwork for pillow placement The copy SOP for the pillows needs some adjustments as well. At first we connect the whiteboxed pillow object with the first slot and the normal SOP with the second input. This is really important here as well as the Template Group. Just add the point group: PointArray_ScatterPositions. The points of this group basically holds all of the centre points of the four subquads in a big tile. We will use these points as roots for the pillows. Leave the flag active at Transform using 'Template Point Attributes'. The whiteboxed pillow was moved to the left which gives us, in combination with this function, the desired result of pillows scattered on the outside wall like a padded cell.



10 Clean up for a test setup To test the complete setup we need to combine both branches of nodes together. Therefore we will create a Merge node and plug both copy SOPs to its input. We prefer to have a clean and structured setup here if we can. At the end, add a null node and call it 'Geometry_Export' for a clean end. To finally complete the room, simply bypass the exploded view node and all of the tiles will fit together again. Just make sure that you don't bypass the Edge Cusp node, otherwise you will have to change the point numbers in the groups.

ZBRUSH

ZBrush

Gustavo Åhlén
facebook.com/
gustavoahlenstudio

As the founder of Svelthe, Gustavo is also a professional 3D/ VFX designer and matte painter

Master ZModeler and Array Mesh

With the release of ZBrush 4R7 came new features such as that of ZModeler and Array Mesh – and these two will be the focus of this tutorial. After learning the next steps, you will be able to model work like you never did before. Before going further we recommend you to learn the fundamentals of ZModeler, then after following a few extra tutorials you will understand how to use each new feature separately.

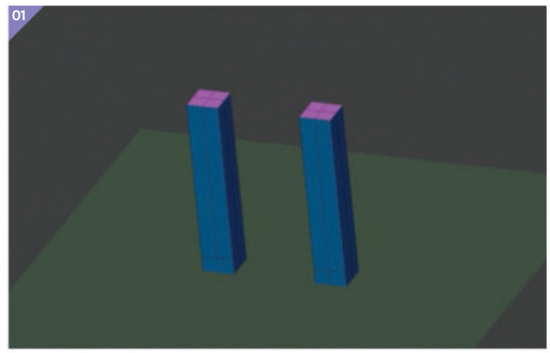
The ZModeler brush has a group of options called Polygon Actions, Target and Modifiers. In Polygon Actions you may notice some familiar options such as bevel, extrude, bridge, split, delete and so on. The Actions group is also linked to Target. To access its Poly menu you need to hover over a polygon before right-clicking. Keep in mind that Polygon Actions operates over polygons, edges, points and curves in a PolyMesh3D. The Target group is fundamental because when you understand how it works correctly, it will save you a lot of time. The Target group describes the polygon selection and to what polygons the actions will be applied. The last group, Modifiers, holds a series of extensive features linked to Polygon actions and this enables us to increase the level of options per action drastically.

We will also be looking at the Array Mesh feature, which will enable us to create infinite copies from a model following different patterns as spirals, circles and so on. Array Mesh is an advanced system that helps you duplicate instances of geometry in patterns and shapes. This system is perfect for creating structures like buildings, geometry patterns and so on, because when you duplicate geometry as an object, it has an Append New option that creates a new stage keeping all the changes previously done. You will notice the previous stages in the option of Transform Stage. Then, when you modify something in the first stage, this will be represented in the last stage.

In this particular tutorial we will create arched doors and columns, and with the help of Array Mesh it will allow any geometry to follow any pattern. The idea behind this tutorial is to show you some of these features with a simple model.

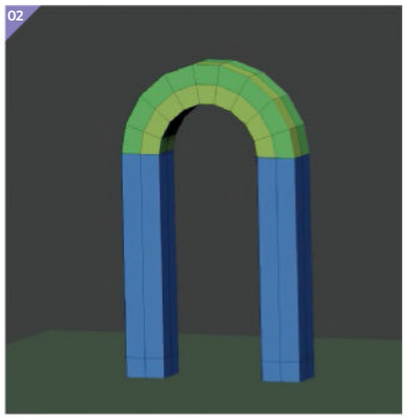
01 Quick mesh with QCube In the first stage you need to add Cube3D, then you need to convert this object to PolyMesh3D. Go to Tool>Initialize and change all the values to 1 for x, y and z, and then click on QCube. Now you will need to hover over the top polygon, right-click and select QCube, set the Target as Flat Island, left-click over this surface and drag, and finally hold Shift and drop. Go to Geometry>Position now and change the x value according to your desired width of the door. Now just go over to the Modify topology and select 'Mirror and Weld', activating the x axis.

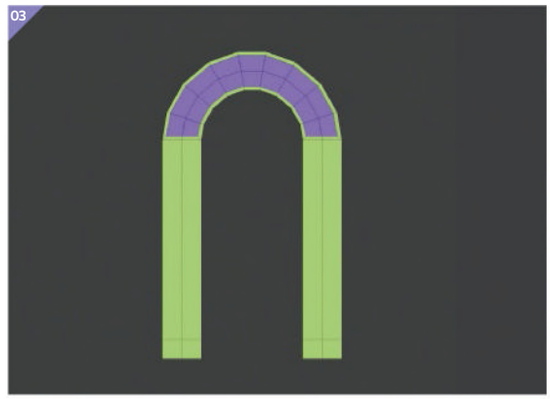
02 Join the parts It is time to activate the x symmetry now by going to Transform>Activate Symmetry. To create the two columns, delete the top polygons of both columns using Polygon Actions and delete. Then, select an edge of the opened top edges, right-click and select Bridge (set Target as two holes, as a circle and then set Modifiers as Optimal Curvature). Once you have selected Bridge, you need to select two edges in the top on both sides and drag and drop. This feature will join both opened columns with a rounded bridge so you can join two separated parts in as few steps as possible.



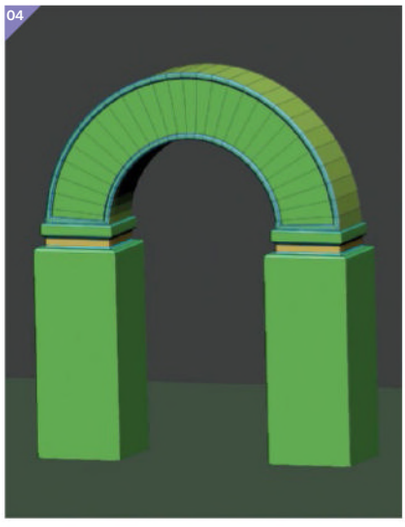
ZBrush ZModeler brush

The ZModeler brush keeps a perfect combination between Actions and Targets. You should test each Action combined with each Target, and then you will understand how to apply each Action correctly. We recommend you learn the fundamentals because it is the only way for you to improve your knowledge in ZModeler. Don't try to create complex things without having previously learned the basics first. Sometimes we try to create some very complex work without trying it out with simple objects first. Once you have made some good progress in basics, this learning process will allow you to improve the complexity in your work later on.

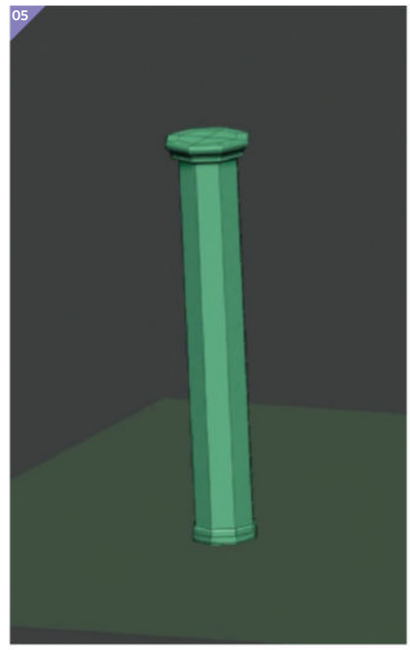




03 Temporary Polygroups ZModeler is a great feature for temporary Polygroups, and to use this feature you need to hold Opt/Alt and select the the polygons that you want to merge into one Polygroup. Select the frontal view of the arch, right-click and select Polygroup (set Target as a single poly) and this will merge the selected polygons into one Polygroup by doing a left-click over the selection. Now, you need to hover over the new Polygroup and select Inset (set Target as a single poly and Modifiers as Inset region) and finally drag and drop.

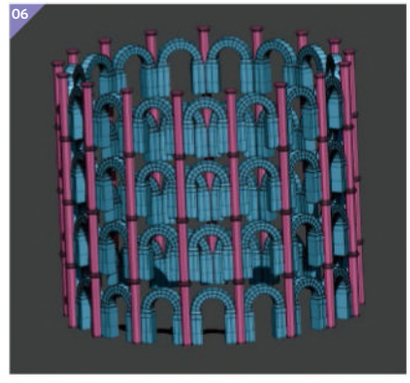


04 Add bevels In this step we extruded some parts of the column, adding new divisions with the help of Insert. Select an edge, right-click and select Insert. Using this option, all the new divisions will be added perpendicular to the edge. If the edge goes across the x axis, the new division will go across the y axis. Once you have added these divisions, you can use Extrude action by selecting a target as PolyLoop. Now it is time to add bevels so just right-click over an edge and select Bevel by using a target of 'EdgeLoop complete'.



05 Make the column It is time to create this column and we need to use the same process as the first step, but in this particular case we used QCyl Y and you can find this in Initialize. Then, using QCube you should follow the same process used in the first step and add new divisions with the help of Insert by selecting an edge. Now you can extrude some areas of the column using a target as PolyLoops to add more details in the columns. Also if you want you can replace this column for a square column.

06 Animate Array Mesh Once we have created the arched door and column, it is time to merge both SubTools into one SubTool, so go to Array Mesh and activate it. Go to LightBox>Array Presets and select the circle preset. Now change the repeat number according to your personal tastes, click on Pivot and move the Z amount until you get all objects in contact. The previous processes created the 'Transform stage 1' and once you get the desired shape, click on Append New and set Repeat as 4. Then activate Offset and change the values of the y axis.



94

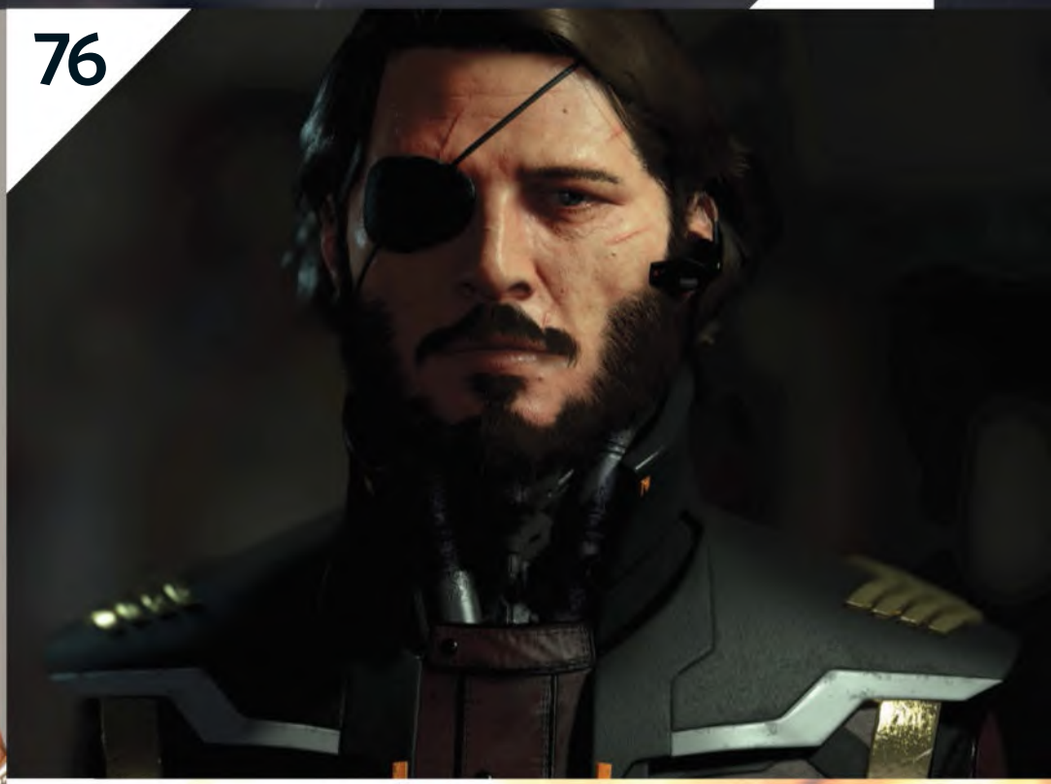
86

68

Texturing


60


78


76


66

> The Edit Stencil tool in Mudbox [warps] the stencils so that fine details will flow around the model convincingly

Resurrect a realistic Velociraptor

Achieve unbelievable textures on an aggressive, dynamic and realistic-looking Velociraptor, using Maya and Mudbox

In this tutorial we will explain the processes of modelling, texturing, posing and rendering a realistic Velociraptor. You will learn techniques for sculpting large muscle groups and skin folds, large, repeating scales and fine details such as small scales, skin pores, wrinkles and bumps. We will cover the texturing process, including how to use the Cavity map to make details pop and use the Edit Stencil tool in Mudbox to warp the stencils so that fine details will flow around the model convincingly. Posing the model is very important and can transform a plain sculpt into something lifelike and believable. Lastly, we will learn how to create materials and lighting in Maya for rendering convincing-looking skin.

JAY HOWSE
Raptor, 2015

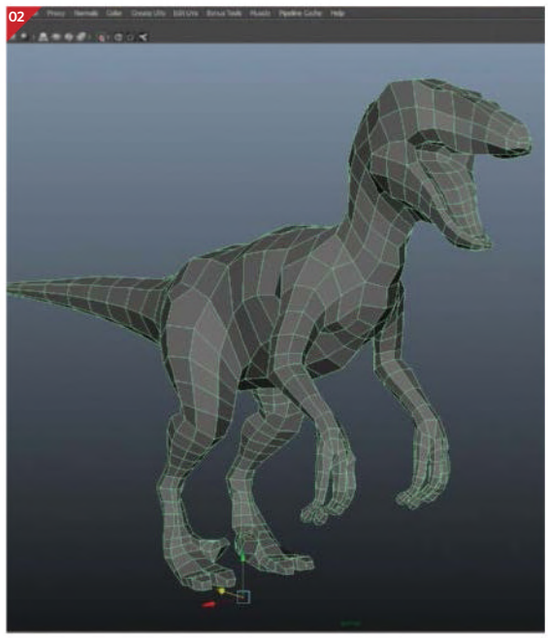
Software
Maya, Mudbox, Photoshop

Learn how to
- Sculpt muscle, skin and scales for the model
- Create textures that are detailed and realistic
- Create custom Vector Displacement Maps
- Create a dynamic pose
- Render convincing lighting and materials

Concept
The concept was to create a realistic and aggressive -looking Raptor from the base model all the way through to a final posed render.

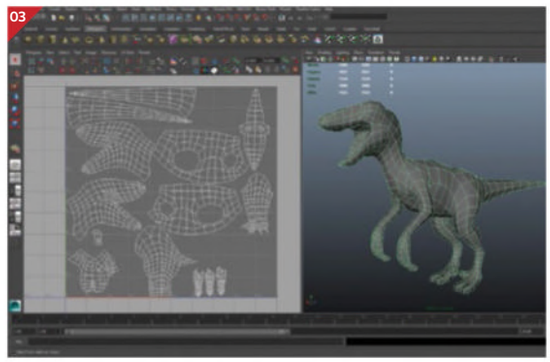
01 Collect references To create a believable creature, the first step is to gather references from real life. We can never know what dinosaurs really looked like, but by observing creatures alive today we can make something convincing. Lizards, crocodiles and snakes are a good reference for scales, skin colour and texture. But, note that dinosaurs were not like modern reptiles (which tend to lie flat with limbs splayed out to the sides), they are dynamic and have powerful hind limbs. Look at bird claws for the feet and even dogs or horses for the underlying muscle and bone structure.

02 Create the base model Set up a side view of your dinosaur as an image plane to work from, ideally from a real skeleton so you can get an idea of where the important bones lie and how they might affect the shape. Model in a symmetrical but relaxed pose with arms and legs slightly bent. This model will be a base mesh for sculpting so it can be quite simple, but do make sure it has the correct number of toes and fingers, and the mouth is modelled in an open position. Leave the teeth, claws and eyes as separate meshes.



04 Sculpt muscle and bone Work in layers as much as possible – this gives more flexibility and lets you try out ideas without worry. Subdivide the model to around 1 million polys, and work quickly and roughly with the Wax brush to build up the underlying muscle, bone structure and overall shape. Use the Move brush for larger adjustments to the shape and proportions, and exaggerate the muscle definition and bones under the skin. Look at the anatomy of dogs and big cats to guide you in areas like the hips and ribcage.

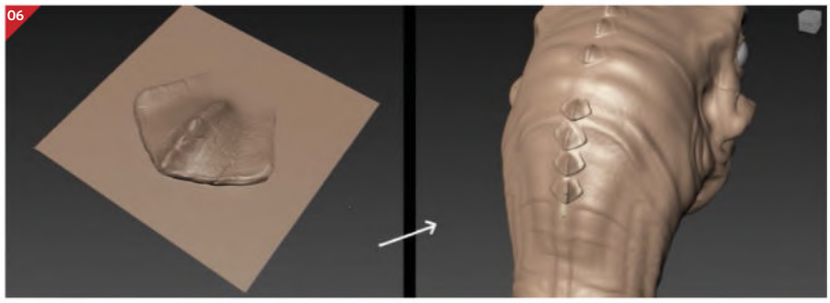
Alternative base modelling methods
In Mudbox you need to start with a polygonal model as a base mesh, but the base model can also be created in different programs. You might like sculpting in ZBrush with DynaMesh or building up your model with ZSpheres, so choose the method which works best for you. One advantage to working with a UV mesh is you can quickly bake out textures and send the model back to Maya. The model can be rigged and posed in Maya and sent back to Mudbox as a sculpt layer.

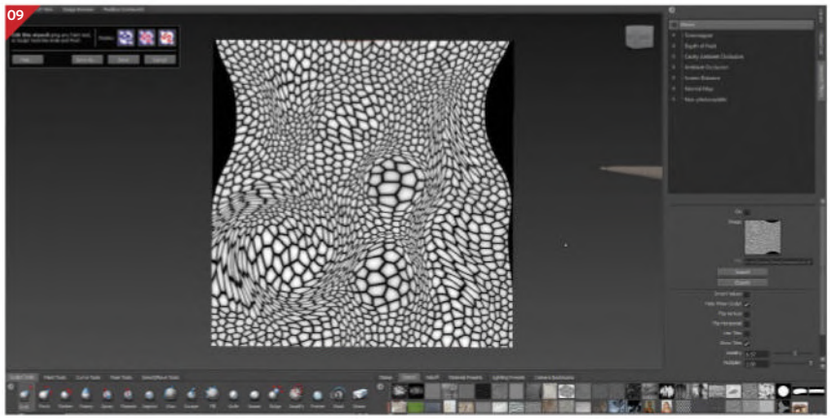


03 Create UVs and clean up Make some UVs for your base model as it will let you quickly try out ideas for texturing and baking out test maps throughout the process. Make sure your model is symmetrical and the geometry is clean before importing to Mudbox and go to Edit>Set Topological axis. This will help if you accidentally start working with mirroring turned off – you can right-click on the layer and mirror the details across. Also, it will enable sculpting with mirroring on a posed, asymmetrical model.

05 Refine the model Smooth out the rough forms and refine the shapes of the muscles with the Smooth and Wax brush to give the impression of skin stretched over muscle. Try to avoid working on fine detail or scales at this point. Use the v_fold_vdm (Vector Displacement Map) brush to create large hanging folds of skin around the neck and limbs. Import and position the claws, eyes and teeth, and sculpt the eyelids, fingertips and gums around the imported meshes (lock the imported meshes so they are not affected as you sculpt).

YOUR FREE DOWNLOADS
from **filesilo.co.uk/bks-798**
- Tutorial screenshots
- Video tutorial



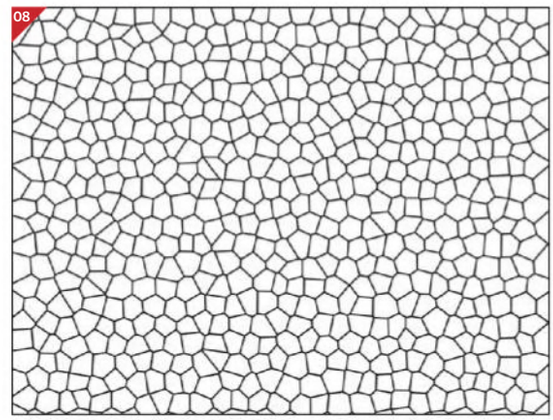


Always think about your render

During the modelling stage it can be useful from time to time to bake out a quick Normal map (even if the sculpt is unfinished) and bring the model into Maya to do some test renders. If you start to set up your render scene early on in the sculpting process it can help to identify any parts which aren't working. Sometimes details will look good in the sculpt, but it can get lost in the more subtle lighting of a render, or the render may show up areas where there is not enough texture resolution or geometry.

06 Add large scales We will create a Vector Displacement Map to make a repeating ridged scale down the creature's back. Create a plane in a new scene, subdivide and sculpt a detailed single scale. Try to create some overlapping parts to add more interest. Create another plane in the same place and use this as the target to bake a Vector Displacement Map. In your scene, import this map as a stamp and use the Repeat brush to create a line of scales down the back, making them progressively smaller as you reach the tail.

07 Sculpt leathery skin Find a hi-res view of an elephant or similar creature from a texture site such as cgtextures.com. Import it into Mudbox as a stencil and using the Edit Stencil tool, move and warp the image into place so it flows with the contours of your model. Use the Foamy brush on a very low strength to brush the texture onto your model. For the neck area the elephant's trunk was used as a stencil, again warping the stencil around the curve of the neck. Use the v_wrinkle_vdm brush to manually deepen some of the cracks.



08 Prepare scale textures Scales are the most time-consuming part of the sculpt, as much of the work has to be tweaked manually for a good result. For a general purpose, randomised hexagonal scale pattern, create a 4K texture in Photoshop and go to Filter>Texture>Stained glass to make a basic texture. This simple texture can be quite versatile as it will be stretched or warped later in Mudbox or Photoshop to create different shapes, but you can build your own scale patterns for different parts of the body if needed.

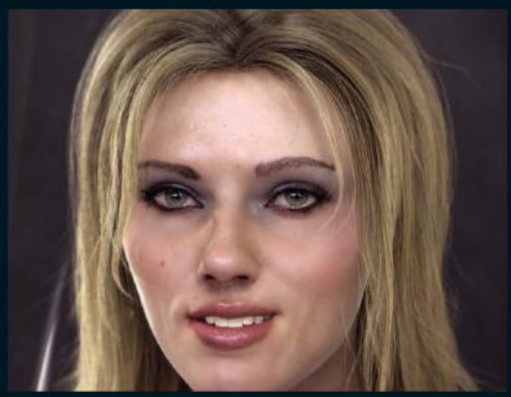
09 Sculpt scales using the stencils Edit the stencil in Mudbox to create bulging areas and transitions from large to small scales. This can also be done in Photoshop with the Liquify tool. With the Foamy brush on a low strength, add the scales to your model, making sure they follow the contours of the body. Use the v_wrinkle stamp to manually deepen some of the gaps and use Erase and Amplify to remove or enhance the occasional scale to add a bit of variation.

10 Add final details Add fine layers of noise and grime to the model to break up the smooth surfaces. Add occasional lumps and pores to the skin, cuts, missing scales, abrasions and so on. Make sure you add asymmetrical details across the central axis like scars, wrinkles and skin folds. Finally check that the fine details are not overwhelming the mid-level shapes of the model. If you've been working with layers you should be able to tweak the layer opacity (you can even push it up above 100 to increase the effect of a layer).

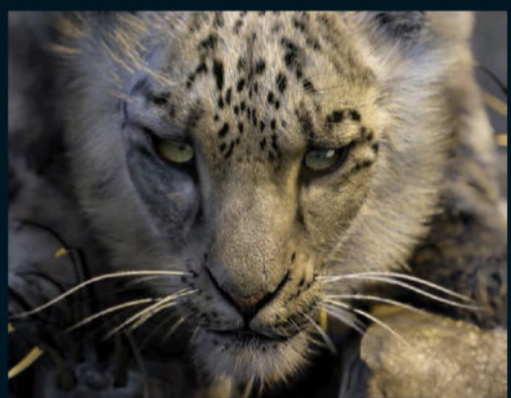
Showcase

Jay Howse

I have worked as a 3D artist in the games industry for around 14 years at Rare and Jagex. I am currently working with Indie studio MakinGames creating all the artwork for *Raging Justice* for Xbox One and PC.

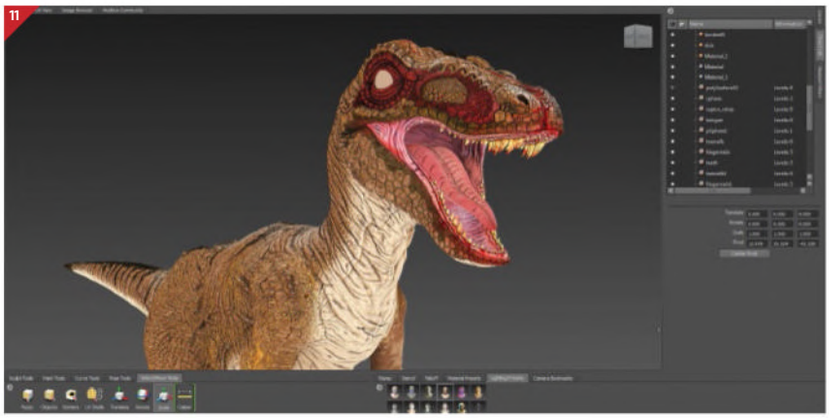


Scarlett **Maya, Mudbox, Photoshop (2014)**
A portrait of actress Scarlett Johansson to help me improve my face sculpting, skin and hair rendering skills.

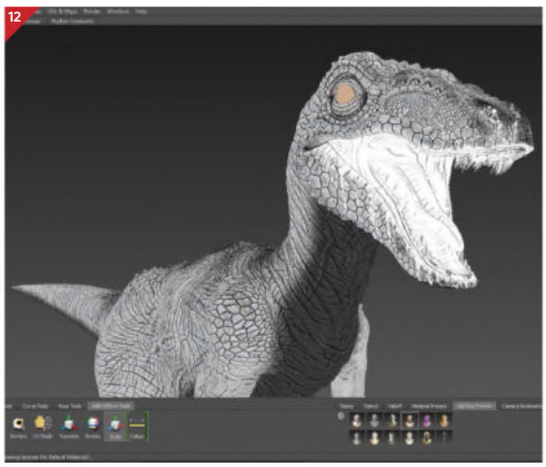


Snow Leopard **Maya, Mudbox, Photoshop (2011)**
The rare snow leopard – sculpted in Mudbox and rendered in Maya/mental ray with Maya fur. Depth of field was achieved in post with a depth map and lens blur in Photoshop.

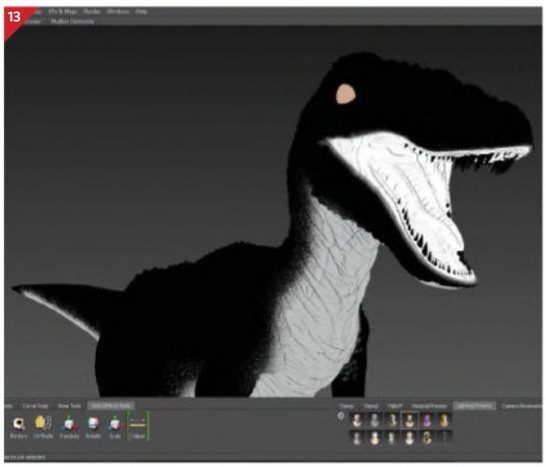
Chimp **Maya, Mudbox, Photoshop (2010)**
A render of a young chimpanzee, I wanted to create the feel of a piece of wildlife photography.



11 Creating Diffuse maps Once the sculpt is finished bake out a Normal, Cavity and Ambient Occlusion map. Use Mudbox to block in key areas of the model on different layers – tongue, gums, mouth, claws, underbelly, eyes, dirt/dust and so on, and export the layers to Photoshop to add more detail. The Cavity map can be used to create dried dust and mud in the deepest cracks – invert the map and adjust the contrast and brightness until only the deep cracks are visible as white lines. Now make this layer additive and adjust the colour to yellow or brown.



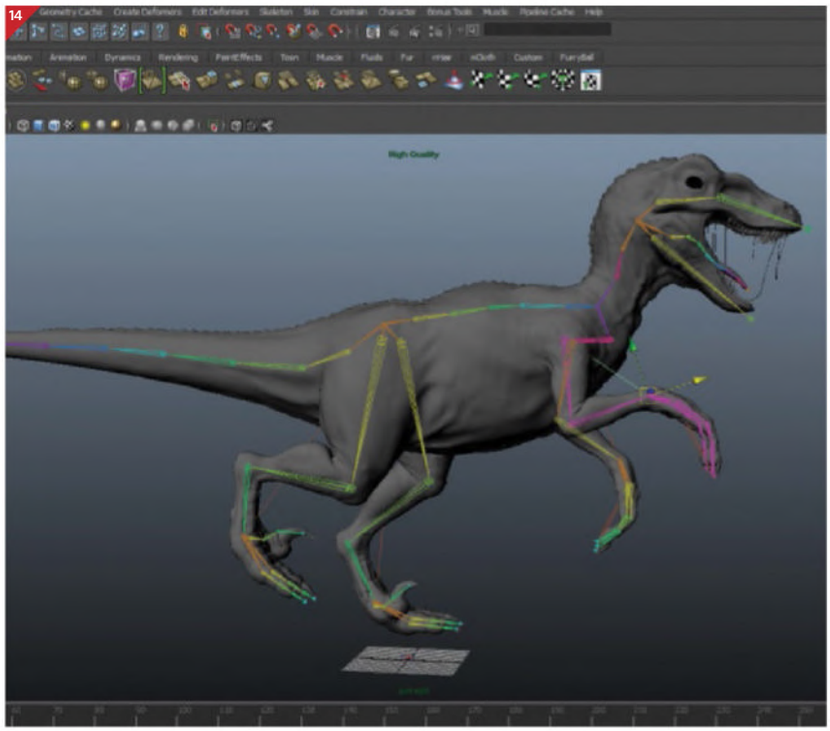
12 Create Gloss maps This map will probably be the one you spend the most time tweaking at render stage, as the ideal values can depend on the lighting in your scene. Start with the Diffuse map and go through each layer deciding how glossy it should be. For this tutorial we wanted the inner mouth very glossy and wet-looking, with drool and specks of blood around the nose and mouth, so these areas were all set to almost white.



13 Subsurface scattering map The mouth area should be really wet, gooey and nasty-looking, so it needs some subsurface scattering and in particular backscattering, but we don't really want any scattering on the hard scales or leathery areas. Use the masks you painted for the mouth, gums and also the underbelly area. The mouth area should be set to white and the belly area maybe a midgrey to let a bit of scattering through.

Displacement and Normal maps

For this tutorial both a Displacement map and Normal map was used. Both maps were baked from subdivision level 3 of the model, which was used in the render. In theory you wouldn't need a Normal map if your displacement has enough subdivisions, but in practice it just seems to help the details pop more. Experiment with this in your own project.

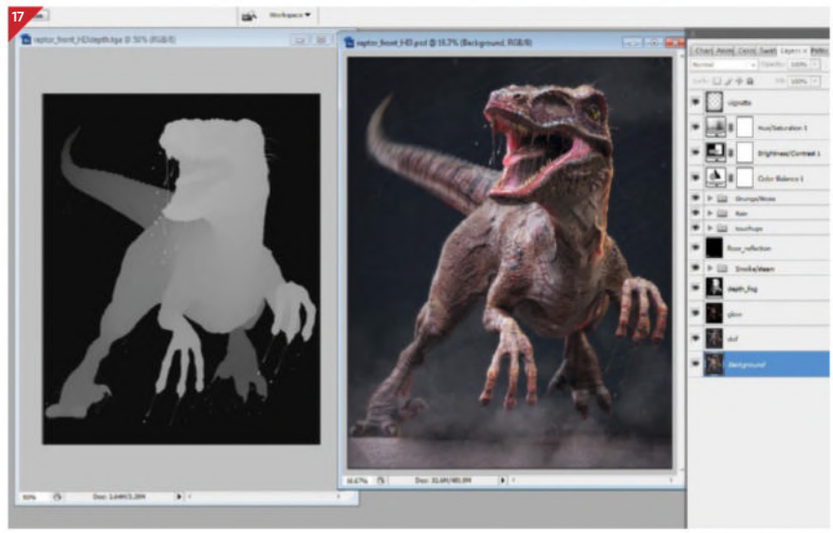


14 Pose the model Export a mid-resolution version of the model from Mudbox into Maya (around 100K-200K) and set up a simple skeleton. It doesn't have to be a fully working rig but if you add IK to the legs and arms, posing will be easier. Skin using a Heat map bind method for best results, and you shouldn't need to do much manual skinning work unless you need really extreme deformations. If required, this model can then be sent back to Mudbox as a sculpt layer to add additional details, flesh folds and so on.

15 Lighting and scene setup We will use mia_material_x materials and HDRI lighting with a linear workflow for realistic results. The easiest way to set this up is to tick 'enable colour management' in Render Settings, then add an mia_exposure_simple node to your camera with gamma 2.2. The important thing to remember is to then set any normal maps to Linear sRGB. Choose a suitable HDR map and add extra directional or area lights if required.

Add subsurface to mia_material_x

To add subsurface scattering to an mia material, create a misss_fast_skin_maya material and set the ambient, diffuse colour and overall spec to 0, then middle-button drag the shader into the Additional Color slot of the mia_material. Finally, in the Shading Group of the mia_material add the lightmap node created by the sss material (under Lightmap Shader, drag in the lightmap node from the Hypershade – it should be called something like misss_fast_lmap_maya). Now the scattering effects from the skin shader should be added to the mia shader.



16 Extra tiny details Create strands of drool and water droplets around the mouth and apply a mia_material_x with the Water preset. Create lots of little spheres in the mouth and on the tongue to simulate bubbles of drool. Apply another mia_material, but this time use the glass_thin preset and change the IOR to 0.33. Bristles were modelled in Maya and added to the neck and back to add a bit more interest and break up the silhouette.

17 Final touch ups and compositiing Render out a depth map and use Lens Blur in Photoshop to add some depth of field effects to the image. The depth map can also be used to create some subtle distance fog. Add layers of smoke, rain and dirt in Photoshop to give the image a bit more life and utilise Color Balance and Curves to tweak the image to your liking for the final image.

SUBSTANCE

Next-gen texturing with Naughty Dog

Substance
Andrew Maximov
artisaverb.info

Andrew works at Naughty Dog, the developer of the *Uncharted* games and *The Last Of Us*.

Seeing a new console generation is an exciting time for everyone involved in the games industry. We may have all been slightly worried about braving the learning curve for next generation art, but it's all cleaner and more structured now.

In this workshop we're going to look at the type of maps you'll need to produce such next-gen art. We'll discuss the types of maps you'll be using in production.

First there's the Albedo map. Albedo is a natural evolution of the Diffuse texture, but they are not the same thing. While Diffuse is just an image skinned to your mesh, Albedo is a physical property that describes the amount of light bouncing off a surface under neutral lighting conditions. This means that Albedo should be devoid of any kind of specific lighting info, like ambient occlusion or specular highlights.

With the specular map, real-time graphics evolution has made it no longer necessary to use the specular highlight approximation system as we have reflections now. Remember that with specularity, what you define in physically based rendering engines is of facing angles only. Everything has Fresnel reflections, so your surface's reflectivity is going to increase the more grazing the viewing angle becomes relative to the surface normal. Now, your engine will use either a specular or metallic map.

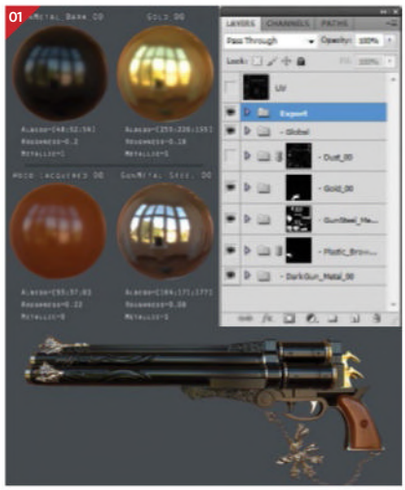
The metallic map is as straightforward as it sounds. Objects made out of metal are white on this texture and dielectrics are black. Grey values are usually reserved for transition between layered materials, for example on dirty metal. When this map is used, the Albedo of your metals is automatically set to black and their Albedo texture info is used as specular.

Roughness/glossiness is the map where you get to let loose the most. On one side of the spectrum, you have a perfectly sharp reflection and on the other an extreme blur. To sum up the differences from previous generation texturing: your Diffuse map is now replaced with an Albedo map, your specular values from now on will have to be more consistent with measurable real-world values and every surface type is capable of reflection, just like in the real world, making glossiness/roughness dramatically more important in describing material qualities.

As next-gen videogame art grows exponentially, the best response is to increase modularity and reuse. Get ready to start building your material pipeline around masks and material presets. The ideal asset production pipeline will look like: Identify Material Types>Load/Create Material Presets>Blend them together>Apply Wear, Tear, Decals and One-off Details.

YOUR FREE DOWNLOADS
from **filesilo.co.uk/bks-798**
- Tutorial screenshots
- Scene files
- Textures

01 Choose your material types for texturing Identifying types of materials you are going to use is the first step in next-gen texture production. You can be very specific and base your material on a real-world counterpart or go with a more generalised approach. It might be helpful to block out the materials with just plain values and see where it takes you. You can then mask out the materials. This is also a great time to block out some of the masks for your material types inside of your PSD, overlay them with your Normal map and preview the blockouts on the actual asset.



02 Create material presets and add adjustments The next logical step would be to transform those rough blockouts into fully fledged tileable material presets. They are going to save you a lot of time in the future when producing assets made of similar materials. In the accompanying image you will see what a source material preset might look like. When these are applied to our spheres, you immediately see that we get some nice material definition. This is also a good time to add any adjustments such as colour tints or roughness variation, especially if you are not producing your material preset from scratch, but rather using one you made earlier. You may notice that the metals share a very similar Normal map, which certainly suits us because after we move things around and tile them around, things won't seem so repetitive. If anything, since all our materials are part of the same object, it is reasonable to expect them to be subject to the same kind of wear.

03 Apply material presets After you've applied the material presets to your model, you'll see that your asset is 80 per cent there and at a distance it could almost look finished. Once you're done with the base materials of your object, all that is left is the wear and tear and whatever unique decals and one-offs you might have.

Dark metals

If you've ever dug into the theory behind physically based renderers you would probably know that there are no dark metals in the real world. Not in their pure form, at least. However, through the history of arms production, humans came up with lots of ways to treat metals to be more durable, while unintentionally altering their aesthetic properties. The dark metal that we're using in this case would have to be blued steel. Bluing is a special process during which a thin layer of magnetite is created on the metallic surface. Magnetite is a mineral with metallic lustre, and that's why it's still going to be white on the metallic texture. Always research your materials before production, especially for guns as there are a lot of surface treatments, such as case hardening, bluing, Cerakoting, Teflon coating and so on.

04 Create wear and tear One of the great tricks that our layered approach lends itself perfectly to is baking the curvature map using the detailed Normal, created by overlaying our preset material's Normals with the global Normal map. Our curvature map, which is a combination of concavity and convexity maps in one, is baked in Substance. This gives you all the scratches and recesses that match perfectly with your base materials and are very easy to lay down. Select all and press Copy, then hit Q for quick masking mode and paste the mask in. Press Q once more and your mask will be transformed into your selection. Using that selection, paint your scratches in where they would make sense. This way you are still very deliberate in your detail placement, but are still using the generated mask and saving yourself a lot of time. This approach once again will get you 80 per cent there, and from then on feel free to refer to custom texture brushes to finish up the painting of the wear and dirt masks.

ALDO VICENTE
Machinist and the Steam Powered Guardian, 2015

Software
Maya, ZBrush, Photoshop, RoadKill, XNormal

Learn how to
- Use 3D character workflows
- Model characters
- Texture and lay out UVs
- Utilise mental ray materials

Concept
I wanted to create a Victorian era machinist and his creation; a giant steam engine, built for military defence. The designs are steampunk-inspired, with a heavy industrial emphasis.

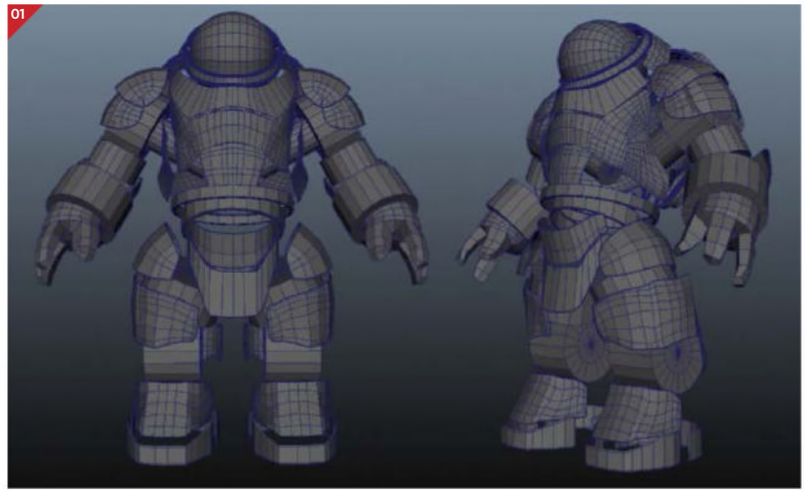
YOUR FREE DOWNLOADS
from **filesilo.co.uk/bks-798**
- Tutorial screenshots
- Video tutorial

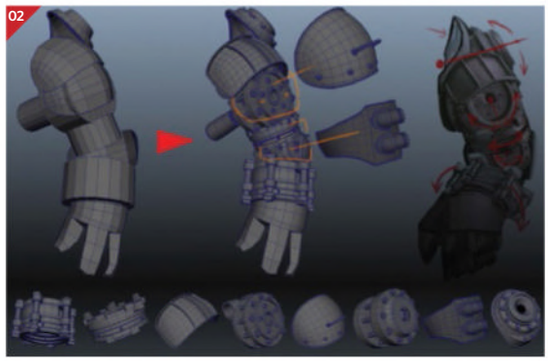
Texture distinct steampunk characters

Master the creation of two technically different characters: a steampunk machine-maker and his giant, steam-powered mechanical marvel

Here we'll be taking an overarching look at some 3D character workflows. We will go through the process of constructing characters, from basic modelling and sculpting to building textures and materials, posing and rendering. We'll be creating a steampunk scene with two characters that are very different from a technical standpoint. Each character will present us with a distinct set of challenges, which enables us to explore a variety of tools, techniques and workflows. While building our machinist, we'll get to practice an organic character workflow. We'll go over techniques for modelling and sculpting anatomy, clothing and accessories, as well as tools for quickly unwrapping clothes and baking perfect detail maps. Our guardian character will have us work through balancing the character's motion functionality and aesthetic appeal. We will also explore techniques for quickly texturing metals and building convincing materials. Finally we'll build a quick environment and set up our scene for rendering.

01 Guardian block-in
Let's start with a quick, rough pass. This block-in gives us an early look at our character's overall shapes and it's important since our character's proportion, overall feel and personality will get lost between the concept art and the 3D model. By blocking in the full character as quickly as possible, with minimal regard for edge flow, neatness and so on, we avoid wasting time on work that will most likely need to be redone. We can worry about edging and cleaning once our block-in feels like our concept.



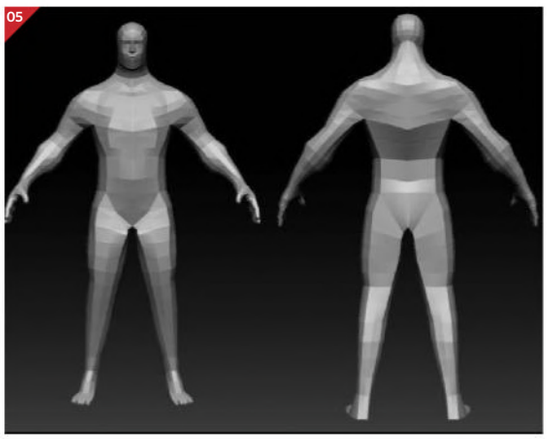


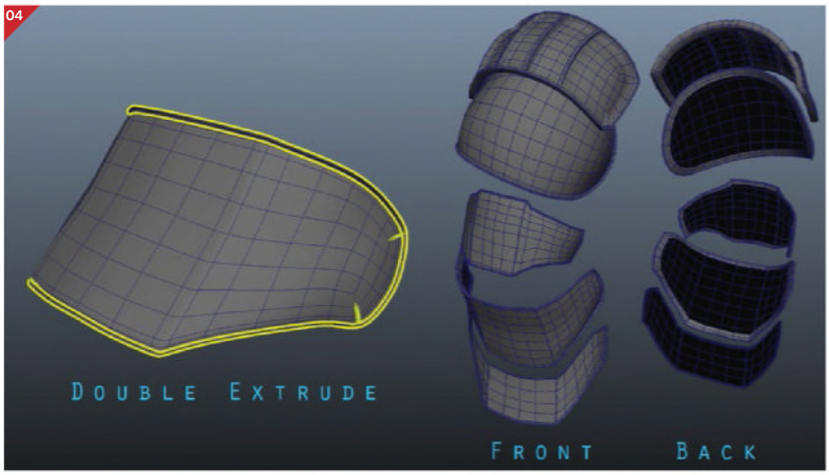
02 Build our parts library
Now we can start looking at our overall forms and figuring out our individual parts based on the character's intended functions. The best part about working on mechanical characters is that, just like in real-life machines, a lot of parts can be reused. We only need to model out a few types of protective plates and one of each part so that the arm can rotate, twist, bend and so on. This means that we are saving a lot of time in modelling and UV layout while creating a cohesive and believable design.

03 Lock down the look
With our small library of parts and our block-in as a reference, we can start finalising our model. We can duplicate, shift and modify existing parts to fill in the rest of our character. We have to keep function in mind to a certain extent (knees have to bend like knees, for example), but there's no need to overthink all of the internal workings of our machine. Form and silhouette should take some priority. Hard surface parts can create very rigid-looking forms, so it's important to emphasise the S-curves in our design to give our character a lively and natural feel.

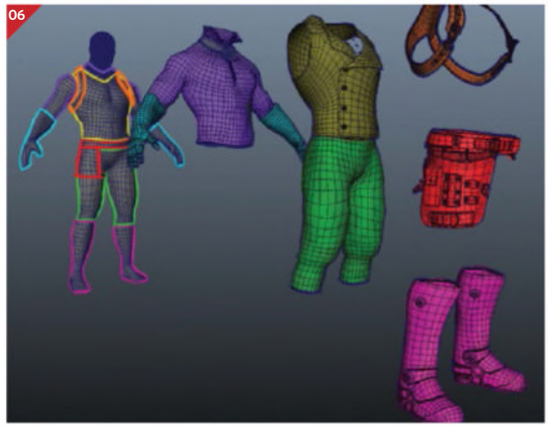
04 Clean our mesh Neat, well-distributed edges are helpful for UV layout and surfacing. The Sculpt Geometry tool on Relax mode, or the Relax Vertex tool are very useful here. Let's delete the back faces on our guardian's outer plates. This will reduce poly count, save UV space and make unwrapping much easier. Double extrude border edges has tons of benefits; we get a consistent edge loop around our mesh, we create support edges for smoothing and UV-relaxing algorithms, and we're adding subdivision density around our borders to provide more fringe detail when sculpting.

05 Build the anatomy In this step we want to create our machinist's anatomy with concept-accurate proportions and clean edge flow. Generally we want to keep this mesh relatively low poly, as we will be using it to extrude our clothing and accessories later on. We can start with an available generic anatomy model, or we can build a simple base mesh in Maya and take it into ZBrush. Here we'll manipulate the physique to fit our character. Our machinist is a cross between a blacksmith and an engineer, so we need his silhouette to convey that he frequently works with heavy metals and equipment.





06 Model clothing and accessories Let's bring our machinist mesh back into Maya, and make sure our character is at real-world scale. Split up our model into the basic areas of our head, shirt, gloves, pants and boots. We can repeat the process to create geometry for the vest, belt and harness. Instantly we have fitting base geometry for our character's costume. We only need to refine our clothes meshes to a certain point, as most of the clothing detail will be done in ZBrush. We can use basic box modelling to shape the belt straps and harness.

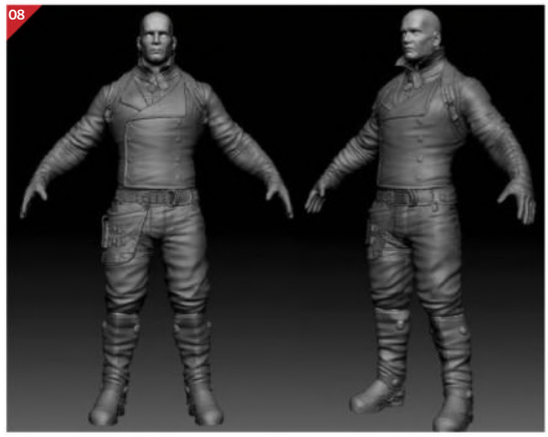


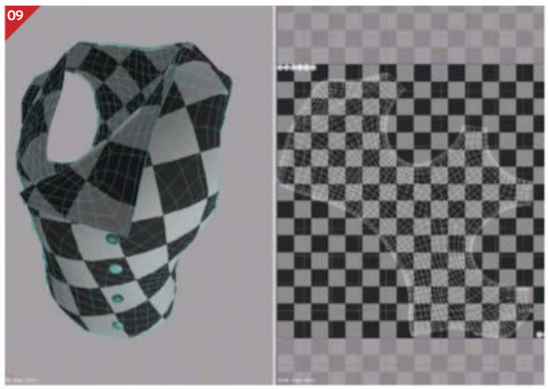
Sculpting warm-up exercise

Sculpting clothes wrinkles can take a lot of practice and iteration. It's about getting in the right mind-set rather than any particular brush or technique. I recommend starting with an exercise to help get a feel for cloth folding. Create a simple sphere on a ground plane, with a plane right above it. Imagine the above plane is a tablecloth, and try sculpting how it would land if you were to drop it onto the sphere on the ground. Try this same exercise with more complex shapes, and imagining different types of fabrics. This helps us warm up and get a really solid grasp of how clothing and folds work.

07 Prepare our costume for sculpting First, we should be sure again that we have neat, even edge distribution. The majority of this should have already been in place from our machinist anatomy model, but we can use the Quad Draw tool in the Modeling Toolkit to quickly and easily make improvements to edge flow wherever necessary. Again we should double extrude our border edges, for the reasons listed in Step 4. We could do UV layouts at this point, but it may be preferable to do it after sculpting as some of the geo may change considerably, and our UVs may need to be redone.

08 Sculpt the clothes Here we'll be sculpting in the large and mid-level detail of our clothing. This includes the overall structure, drape of our clothes and the wrinkles. We won't be sculpting in the fine detail such as the threading and stitching of the fabric, as we'll be adding those details via our textures later. It's good to have reference of the types of fabric each piece is made of. Keep in mind that these are just panels of fabric falling over simple anatomical shapes. Also consider that these folds need to stay generic enough to work in most poses.





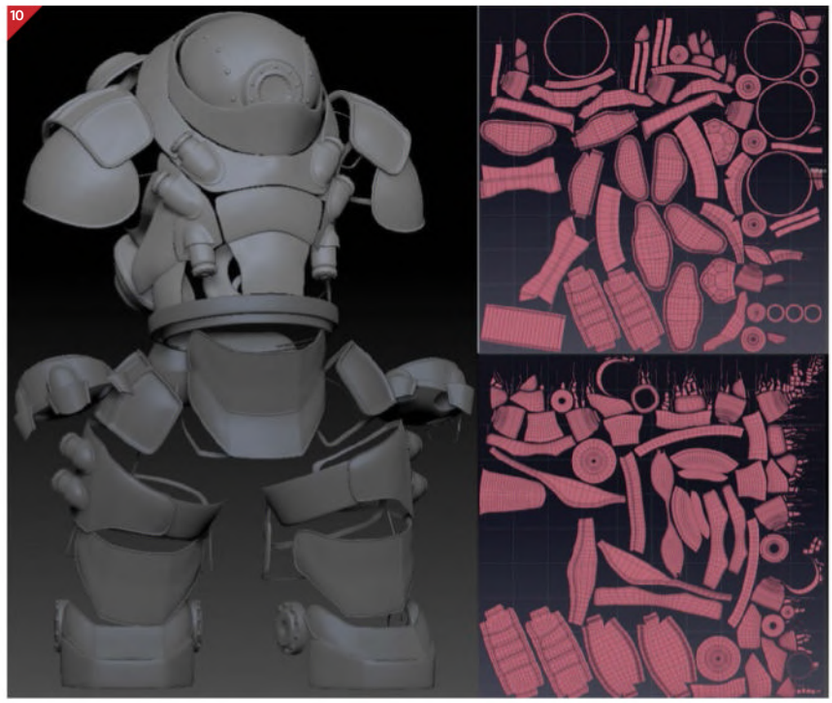
09 Lay out the UVs To unwrap our clothes, we'll use RoadKill, a free plugin for Maya. We can select our vest mesh, and run the plugin. In RoadKill we double-click to select the shoulder seam edges and hit C to cut the UVs. Instantly, we will see our vest flattened neatly in our UV viewer. We can quickly repeat this to separate the lapels and we're done. If we want to smooth our UVs out further, headus is another free option with an excellent UV-relaxing algorithm and a very simple, but fantastic GUI for flagging, bunching and stretching in your UVs.

Image-based lighting

Metal materials are mostly reflective, so in order to get realistic-looking metals, we need to provide them with an environment to reflect. For this, we'll use image-based lighting. In the Render Settings, go to the Indirect Lighting tab, and click 'Create Image based Lighting' to put an IBL sphere in the scene. We can load an HDR image into our IBL sphere node – free HDR images are available online. Now all we need to do is turn on Final Gather. At render, mental ray will use data from our HDR image to light the scene and create some convincing environment reflections for us.

11 Paint metal grunge It's quite hard to distinguish which UV islands correspond to what plates just by simply looking at the flattened mesh. This makes it difficult to paint our textures in 2D. Instead, let's turn on ZBrush Polypaint and subdivide our mesh into the millions. We can start this off by coating our mesh in 50 per cent grey. Let's set our brush to RGB Color Spray with a grungy alpha and zero z-intensity. We'll paint in the edge grunge, rusting and scratches. At this point we are only concerned with colouring this detail onto our mesh. Grunge and rust should be darker values while scratches should be a white colour.

12 Create the metal textures Now we should export our Polypaint data as a texture map and bring it into Photoshop as an overlay layer, with our base metal colour underneath. The grey in our grunge map will now disappear, leaving us some dark grunge and light scratches that will go over our base colour. We'll also add a subtle metal photo texture over our base colour and colourise our grunge to get a rusty red hue. This is our Diffuse map. We'll fashion a Bump map with desaturation and then invert our scratches to black. We'll raise our base colour to near white and then adjust the noise layer to a medium grey.

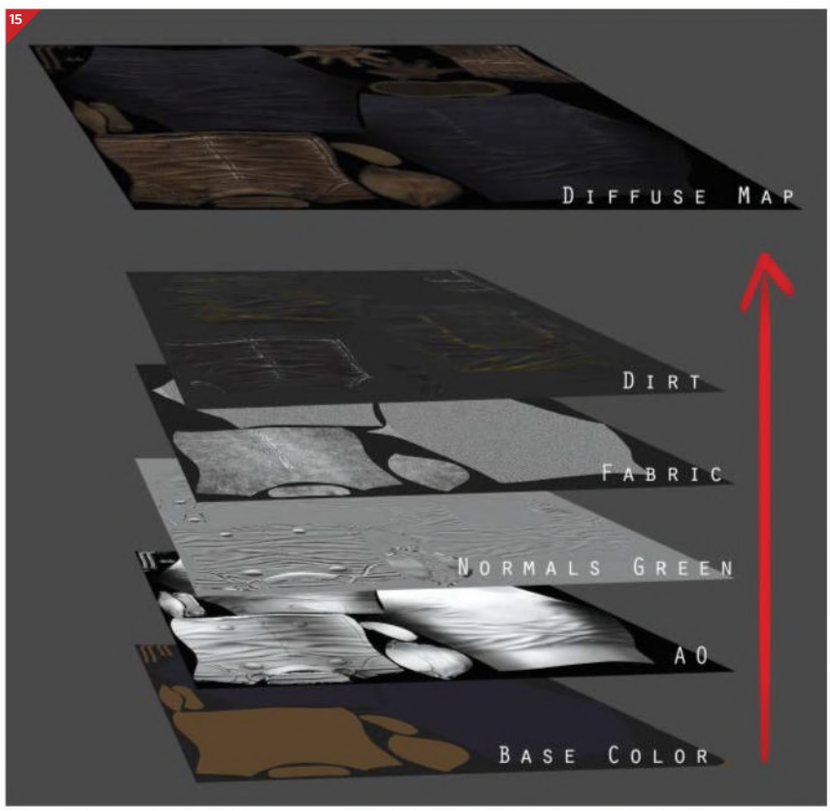


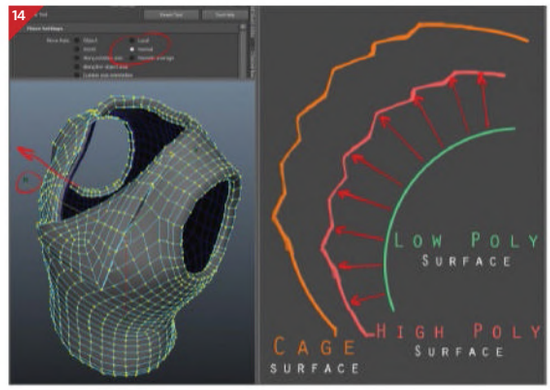
10 Unwrap the guardian Our guardian has a lot of internal and external parts, but we can save a lot of time and UV space by using 3D procedural textures on the smaller, less visible internal parts. We'll only unwrap the outer metal plating of our guardian. Let's select our plated parts, export them to ZBrush and run the UV Master plugin. Because of the cleanup and double extrudes back in Step 4, our plates are essentially curved planes with structured border edges. This makes them ideal for the UV Master algorithm. Almost instantly, we get a clean, evenly scaled UV layout for all our plates.

13 Create metal materials Metal materials are reflection-based so we'll use image-based lighting to give our metals something to reflect. Using mia_material_x_passes, we'll plug in our Diffuse and Bump maps. Metals have low diffuse weights. Our metal is relatively rough – it has a smooth, metallic microsurface, but its overall surface is too rough to cleanly reflect any image. We will get a certain level of gloss but our specular reflectivity should stay low. To control this further, we can fashion a Specular map from our Bump map by adjusting the levels to fit within our desired reflection values.

Understanding map baking in XNormal

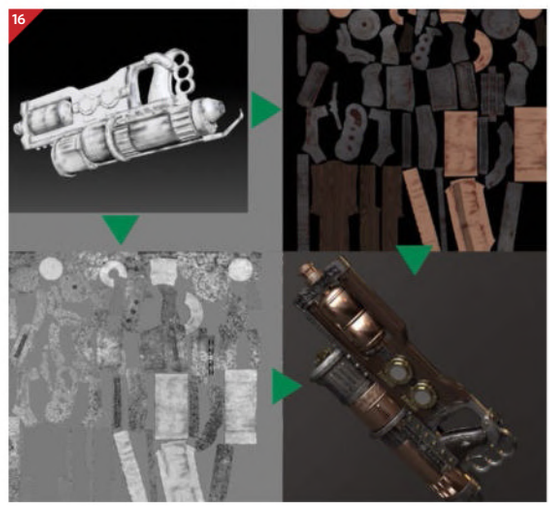
XNormal casts rays from the lower-res mesh surface to the hi-res surface. The rays' travel distance and normal data is written as colour data to the UVs of the lower-res mesh to create our maps. However, there is a travel distance threshold – if the rays travel past that distance without reaching the hi-res surface, they get ignored and no data is written to the map. This is why we use cages around our meshes instead of distance thresholds. The cage surface becomes the new distance threshold, ensuring we collect all the hi-res data, with optimised calculation time.





14 Bake good maps We'll bake our maps using XNormal. With our vest as an example, we'll export the high and lower-res meshes. To create a cage mesh in Maya, we'll duplicate the lower-res mesh, and with Translate set to Normal we'll move all the verts out along N. The cage mesh should be just big enough to fully envelope both high and low-res meshes. Now we'll take our meshes into XNormal. Right-click on the low-res mesh and click 'Browse external cage file' to assign our cage mesh. Set to Average Normals and Bake! This will produce full, clean, accurate and fast maps!

15 Texture the machinist Normal and AO maps are a great starting point for Diffuse, Bump and Specular textures. In Photoshop, paint base colours for each island in our UV map. Multiply the AO map over the base and Overlay the green channel of our Normal map to create top-down shading. Now we'll tile and overlay images of fine fabric detail. We can use those same overlays later on top of a grey background to create our fabric bump texture. Let's paint in some subtle layers of dirt, dust and grunge to finish our diffuse texture.



16 Texture the equipment For our equipment, textures, we'll use a technique similar to Step 11. With the help of ZBrush Polypaint, we'll colour in the rusting and scratching around our equipment's nooks and edges. We can bring these maps into Photoshop and colourise or do a photo texture to create a grunge effect. Again we'll repurpose the same maps to create our bumps, indenting scratches, dinks, rust and noise. We can add neat decorative pieces by adding black-and-white patterns into a Bump map, and overlay these patterns into our Diffuse map to make them pop even more.

17 Clothes shaders Let's import our machinist textures and plug them into a new mia_material_x_pass. Remember that the Normal map gives us the fold and wrinkle detail, while our fabric bump maps gives us the fine stitching and cloth detail. Both will be plugged into the shader. The cloth for our vest, pants and shirt is generally pretty diffuse, with very low gloss and specular reflectivity. We should also adjust the BRDF as looking at cloth at direct angles is especially dull. We can add a tiny bit of translucency, or we can fake it by plugging in a very weak mia_self_illumination node.

18 Build the remaining materials We'll need to build several different material types for our equipment. With our maps ready, gather reference and create some fresh MIA materials. First, name all our materials, turn on Use Fresnel, and input the real-world Fresnel values for each one. Now we can start balancing the Specularity and Diffuse, knowing that our reflective distribution is accurate. Since there's no refraction, we can observe our reference and estimate the reflective strength, and the shade strength from zero to one. With MIA's energy conservation math in mind, we can solve for really accurate reflective and diffuse weight values.

> "With the help of ZBrush Polypaint, we'll colour in the rusting and scratching around our equipment's nooks and edges"

MIA energy conservation math

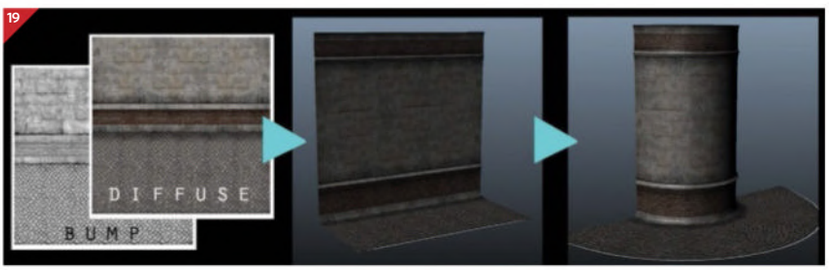
The MIA materials are physically accurate, meaning that Diffuse plus Reflection plus Refraction will never be more than one – the total incoming light. But how do our input values get balance out? When we increase the value of one of these, its resulting effect takes energy from the result of the others to maintain the same total energy output. It breaks down as follows:

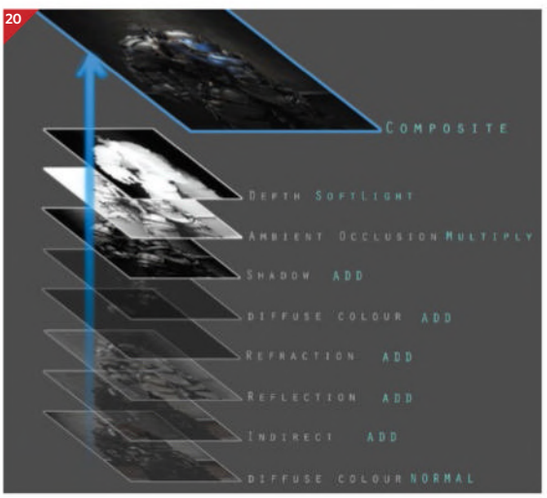
Diffuse result = (Diffuse weight value) – (1/2 Reflectivity attribute value) – (1/1 Transparency value);

Reflectivity result = (Reflectivity attribute value) – (1/2 Diffuse effect) – (1/2 Transparency attribute); and

Refraction result = (Transparency attribute value) – (1/2 Reflectivity attribute value).

19 Make a background environment Let's create a simple ground or wall backdrop. We'll give the wall extra edges to indicate where the brick meets the concrete segment. We can use planar mapping to quickly get UVs. In Photoshop, we'll use photo textures of bricks and concrete to create a quick Diffuse map. We'll balance the levels and colours to make the different images fit each other. Now we can fashion a Bump map by desaturating and pushing recesses into black, while pulling convex surface space up to white. To add depth to our environment in Maya, we can apply a bend deformer and create a rounded wall.





20 Render our scene Let's set up render passes for better control over our final image. In Render Settings>Passes, we'll create and associate beauty, AO, diffuse, Indirect, reflection, refraction, shadow and specular passes to all three. We can render and save these passes out individually for compositing in Photoshop. We should use 'Linear Dodge (Add)' blending mode to put our passes together and re-create our beauty pass. Finally, let's create a custom matte pass, assigning different coloured surface shaders to each character and also to the environment. Now we have full control for tweaking each asset and element to make our final image.

ZBRUSH, MAYA

ZBrush, Maya

Tom Parker
tomparkersartdump.wordpress.com/gallery

Tom has been working in the games industry for over six years and worked for studios too. He is now a freelance character artist, working with clients from all over the world

YOUR FREE DOWNLOADS
from **filesilo.co.uk/bks-798**
• Tutorial screenshots

Create realistic scarring

Scars are often used as a storytelling device in most character designs. They get you to question everything about the character. What events led up to them getting that gruesome thing? What did they do to deserve it? What personality might they have? So it is important to think about what you are doing and if your character even needs scars in the first place. As we are going for realism we will need to make sure we can answer these questions, and more, to add depth and believability to our character. We are going to make our scars using ZBrush, Maya and Marmoset's Toolbag 2.

Scars can originate from many different injuries or birth defects, so we need to make sure we know what we are after before beginning. In this tutorial we will concentrate on concave and convex scarring techniques, which can be applicable to most tissue scarring including burns, birthmarks and stitching. The techniques are very simple and will take no time at all.

We will start by doing a bit of research and considering their placement on the model. This is probably the most crucial part of the tutorial – if the placement is not correct it can ruin the entire realism of the character. The scars will have a large impact on the overall design so we will need to be careful and be very selective of their placement.

Then we will move onto sculpting. We can add scars at any time during this phase. If it's smaller scarring, add them close to the end so that you are not distracted from the forms that you are trying to find in the face whilst sculpting. But if they are significant scars (hero elements) that are important to the design, then it may be important to get them in early and then work around them. Every character is different so you will need to use your best judgement.

Next we will prep them for texturing before taking the model into Toolbag 2. This is where we will be able to view our results in real-time and allow for quick iterations when needed. Let's begin!

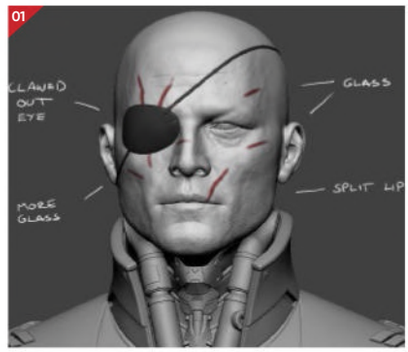
01 Decide on the scar placement and position The actual placement of your scars will play a large part in the character's believability and will impact your design considerably. First thing to do is to take a few minutes and decide where you want to place your scars, also why they are there, and how they were made. This will all impact the type of scarring you need to replicate, help add depth and stability to your character, and minimise random placement as much as possible. For this character, we wanted to accentuate the scarring stereotype so we placed many scars all over his face.

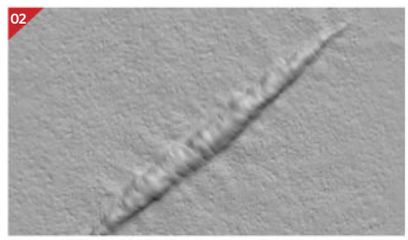
02 Sculpt convex scarring Now that we've decided on what kind of scars we want, we can start sculpting them in. For the convex scarring, pull the healing tissue out by using a quick stroke with the Standard brush. Then by using the Inflate brush, we can add some swelling, which will add to the fleshy feel. If you have the resolution you can also add a few tension creases with the Dam Standard brush on a low intensity. We can also add some surface detailing with a Spray brush and small Alpha.

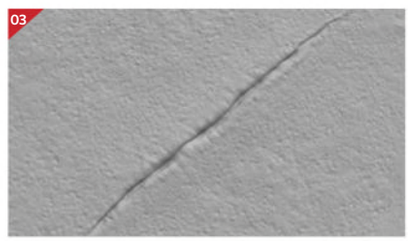
03 Sculpt concave scarring For the concave scars that cut into the skin we simply carve into the skin with either a Standard or a Dam Standard brush. Scars generally do not heal straight so try to add a touch of misdirection here and there. You also often get a bit of pull on the tissue surrounding these, so using the Inflate brush we can add some subtle distortion. Now we can add surface detailing to this in the same manner as the previous step. Adding this extra detail will really help us when it comes to the eventual baking process.

04 Polypaint the skin When we are happy with our sculpt and we are ready for texturing we can start Polypainting a first pass for our Albedo/Diffuse maps. This we will use as a starting point for our textures. So we first start by Polypainting the skin with a Spray brush, adding in subtle hues here and there and skin blemishes. Now we need a nice fleshy colour for the scars. We can then bake out the Polypaint information and start refining our textures in our painting software.

05 Create textures Making believable real-time skin in Toolbag 2 is very easy due to the supplied skin shader. We will need a Normal map, Albedo map, Gloss map, Subdermis map and a weight map to get the results that we want for our character. Marmoset's website has a fantastic video that can explain working with their Skin Shader and all the little tricks used in creating realistic skin.







Research is key

Make sure to do plenty of research into whatever you are doing as an artist. It helps re-enforce the realism and will ensure better results than just working from inside your head. Scars can be a bit gruesome sometimes so just be wary when searching online. With skin we are lucky as we all have it, so make sure to observe yourself in the mirror and stare at others to see how different peoples' skin reacts to light. Watching movies is a good way of observing skin too, making replicating it easier.

When creating realistic skin, it is important to work from reference and try to replicate the results the best you can. Lighting will also play a huge part in how your skin shader reacts.

06 Make the scars pop Now that we have a nice skin shader, we can start tweaking the maps to make our scars pop. Scar tissue is usually a lot smoother than standard skin, so to simulate this we can edit our Gloss map by making the scar tissue a touch brighter. The next thing we do is tweak our Subdermis map by using a deep red on the scar tissue to give a more translucent and fleshy feel to them. You can also add the scars to your weight map to enhance the translucency effect.

Model and texture a Roman Centurion

Bring a detailed legend to life in just a few steps

Over the next 17 steps we will take you through a typical workflow for designing and creating this type of epic character. This tutorial was written in a way so that it can be followed by anyone with at least a basic understanding of 3D software. We will start off in ZBrush 4R7 to build the character, switch over to Marvelous Designer 4 for cloth and create weapons in 3ds Max.

Finally, we'll finish up by showing you how to quickly create some realistic renders with KeyShot 5 via the new Bridge feature.

> Starting off with a base mesh, subdivide a few times and build the head using mostly the Clay brush and Standard brush

DAMON WOODS
The Centurion, 2015

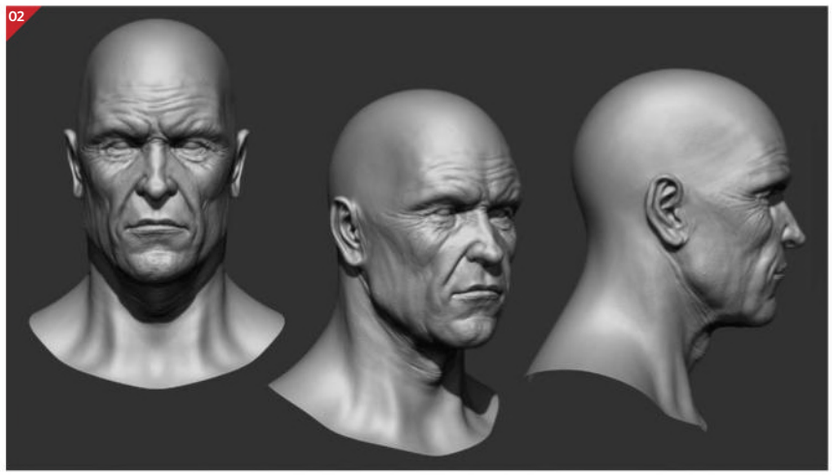
Software
ZBrush, 3ds Max, Marvelous Designer, KeyShot

Learn how to
- Mask armour in ZBrush
- Create ornate armour designs
- Make realistic drapery using Marvelous Designer
- Create custom insert brushes in ZBrush
- Construct realistic hair using ZBrush FiberMesh
- Easily render photorealistic images using KeyShot 5

Concept
My inspiration came from the amazing armour designs that I've viewed in museums from various empires in history.

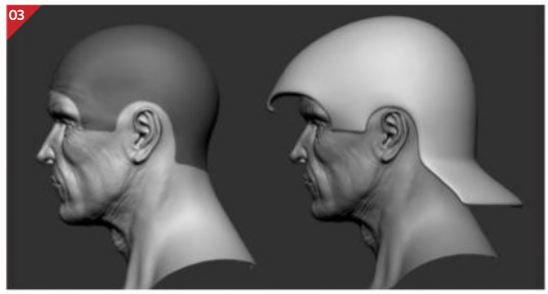
YOUR FREE DOWNLOADS from **filesilo.co.uk/bks-798**
- Scene files
- Tutorial screenshots

01 Gather good references Before we start up any programs, spend a good amount of time gathering references as it gives you a chance to really think about where you're going to take the character. The internet is a great place to start, but don't limit yourself to just one source. Being an artist means treating anything and everything as a potential source of inspiration. (Yes, sometimes that even requires going outside!) Museums, books, toys... it's all at your disposal. This piece was heavily inspired by legendary renaissance armour maker Filippo Negroli so we recommend you look him up for armour reference.

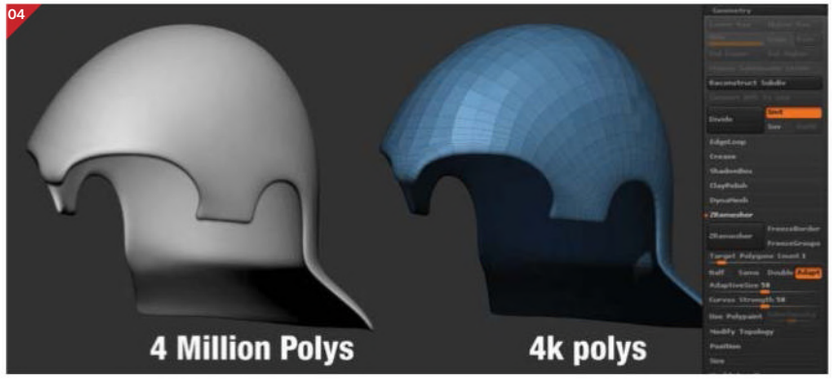


02 Sculpt the head Starting off with a base mesh, subdivide a few times and build the head using mostly the Clay brush and Standard brush. Research has taught us that soldiers weren't even allowed to reach Centurion status until at least the age of 30. So we've chosen to go with an older looking man. We assumed they spent a lot of time in the sun so we focused on giving him some lines around the eyes and jaw areas (Clint Eastwood was an inspiration for that).

The power of DynaMesh
DynaMesh is possibly the most useful tool in all of ZBrush. It lets you turn any object into an infinitely flexible ball of clay. Simply go to the Geometry tab and hit the DynaMesh button so that it turns orange. As long as it's activated, you can push and pull geometry as much as you want and simply Cmd/Ctrl+drag in the workspace to fix any stretching.



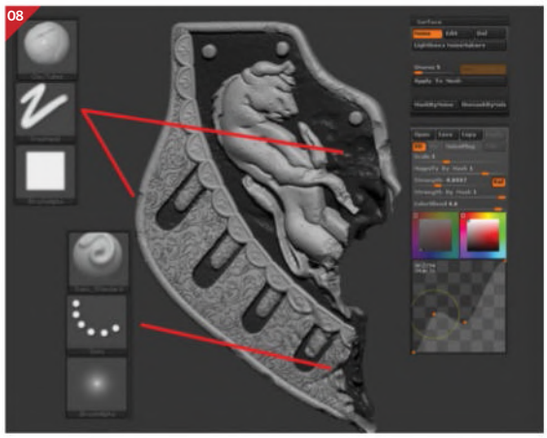
03 Create the helmet shape The first step in creating a helmet is to ensure that you get a good base shape that you can work with. We are going to use Mask and Extrude to achieve this shape. Once you've got the base shape, scale it up slightly and DynaMesh it so that you can push and pull it into the shape that you want it to be.



04 Apply ZRemesher When you have the shape you want, hit it with ZRemesher with the res set to about 1. This serves a few purposes. First, it gives you a nice clean low-poly version that you can always down res to when you want to save memory. Second, low-poly meshes play much nicer with programs like Transpose Master and GoZ.

05 Create Alphas We've used a wide variety of Alphas on this piece to create all of the ornate armour decoration. There are a ton of royalty-free ornate swirls and designs online to choose from. Just make sure that whatever design that you end up using is one-to-one (in a square box) and that the image is fully black and white to avoid unwanted noise or distortion.

06 Apply the Alphas With the Cmd/Ctrl key held down, switch the draw mode to Drag Rect and the focal shift to -100. Then go into the Alpha slot and load in the custom Alpha. Holding Cmd/Ctrl again, select the Alpha and drag it onto the surface of the helmet. Then do an Extract to create the helmet. You'll notice that the extraction only gives you the overall shape of the Alpha and ignores smaller details. With the piece that you just extracted, you'll notice the mask is still applied with all of the detailing still inside. Now use Inflate and Standard set at lower intensities to bring out the smaller details. Repeat this for the whole helmet with different Alphas.



07 Use the Insert Mesh Brush To create the ornate borders around the edges, create an Insert Mesh brush. To do this, create a simple repeatable shape – in this case, the two small leaf shapes. Make sure to keep them as low poly as possible because they will be repeated many times. Point the camera at them from the angle that you want them to lay on the surface. Then, in the brush pallet, hit Create Insert Mesh and then choose New. With the Insert brush selected, go to the Stroke tab>Curve and turn on Curve Mode. For this particular brush, set Curve Step to 0.6. Then just draw the ornate design along the edge.

08 Weathering and battle damage
No seasoned warrior is going to have spotless armour. These soldiers had to endure the harshest weather and countless attacks. Some well-placed damage can add a lot of appeal and realism to a character. Use the ClayTubes brush set to subtract to add chipping dents. The Standard brush with Spray mode and Alpha 08 are great for adding surface irregularity. Finally, apply a bit of noise to everything for good measure.

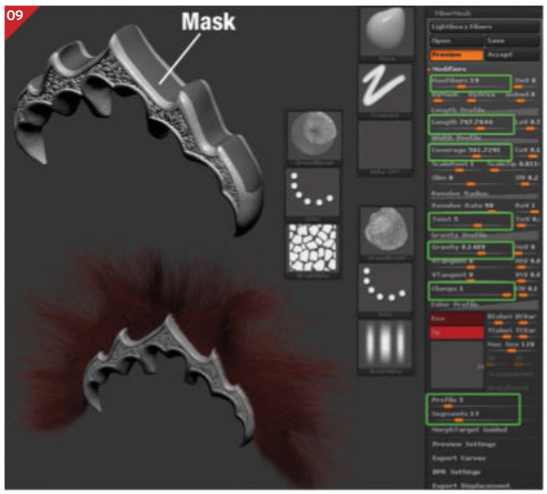
Showcase

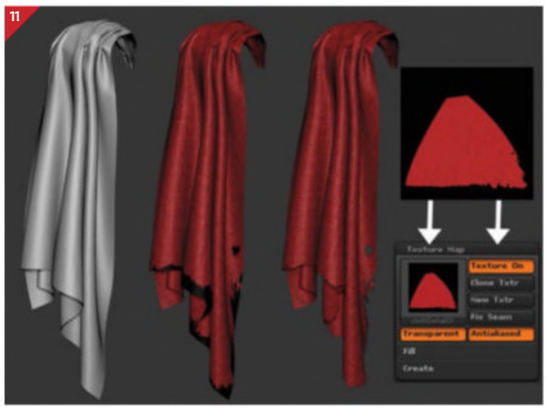
Damon Woods

There's nothing I'd rather be doing than what I'm doing right now. Being a part of this cool little industry gives me a chance to be a part of the magic that entertained and inspired so many over the decades.

Queen of the Sea **ZBrush, Photoshop (2013)**
I wanted to convey a female character who is both strong and unconventionally beautiful. The main goal was to create a believable person who looked like she had experienced a lot of pain in her life and somehow came out on top.

Zombie **ZBrush, Keyshot (2014)**
A ZBrush sculpt that I did after a weekend zombie movie marathon. This decaying monster probably wasn't too nice of a guy even before he was bitten. This was my first piece rendering in KeyShot as well.





11 Detail the cloth Using the UVs generated from the cloth, add some details in Photoshop. Paint in some edge tearing with pure black, and add some cloth texture as well. Then load the texture from the texture pallet in ZBrush, and turn on Transparent and Antialiasing, which turns all pure black values to transparency.

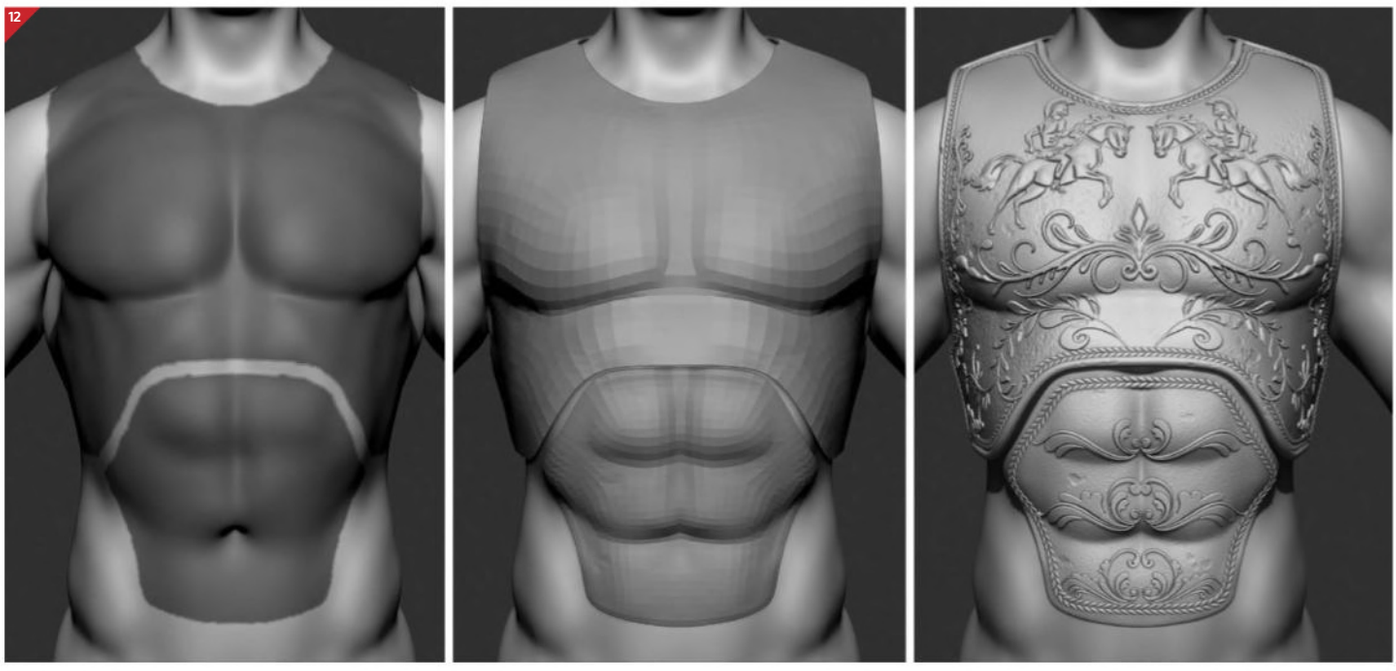
09 Create hair with FiberMesh Create the top head piece from a DynaMeshed sphere, then mask the area that you want the hair to sprout from and apply FiberMesh. After that, groom it a bit so that it doesn't look so perfect and new. Once again, embrace any imperfections in your work.

10 Create the cloth All cloth elements were created in Marvelous Designer and then imported into ZBrush. For the undershirt, it has to be pretty much skin tight to avoid penetrating the form-fitting armour. Once you have the basic shape, import to ZBrush and sculpt the smaller folds by hand. The neck scarf and cape are very simple shapes with a few carefully placed pins. Marvelous is such a great tool for creating realistic cloth and another great benefit is that it automatically generates UVs as well. This really helps out in the next step.

Alternative hair option

There are a lot of great programs out there for creating hair. If FiberMesh isn't getting the results you want, try using 3ds Max's Hair And Fur modifier. Instead of masking the area that you want to generate hair from, you can export the piece using GoZ, duplicate the faces in Max and then add a Hair And Fur modifier to it. There are a lot of great tools for styling the hair there. When you're done, simply convert the hair to geo with Tools>Convert>Hair>Mesh and send it back to ZBrush.

12 Create the body armour Just like with the helmet, use the Mask and Extract method to create the armour and to make molded body armour that supports the anatomy underneath. Use Trim Dynamic and sPolish to refine the shapes after you extract. When creating the ornate designs, choose shapes that would further support the anatomy of the soldier.

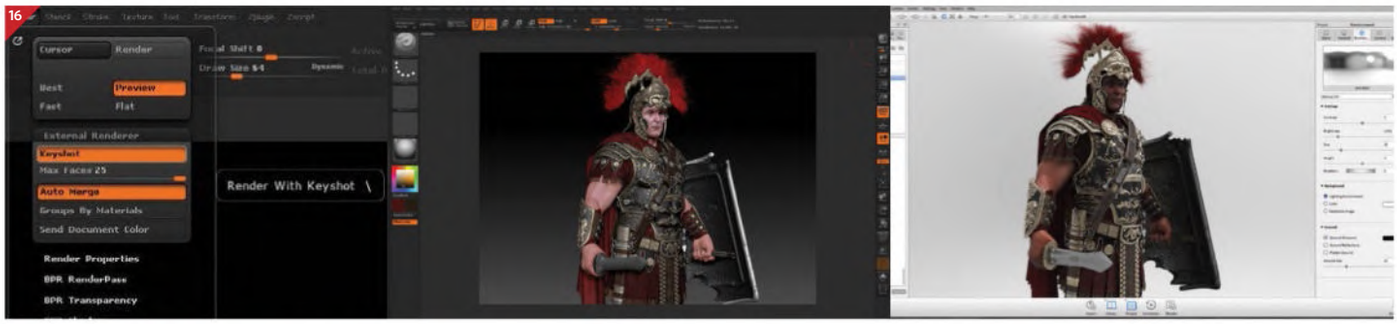


"Marvelous Designer is such a great tool for creating realistic cloth and another great benefit is that it automatically generates UVs as well"

13 Create the weapons What's a soldier without his weapons? This is one of the most defining aspects of a warrior from any era. Artistic licence aside, it's very important to do your homework on the type of weapon that goes with your hero. You wouldn't equip a Viking with a katana or a samurai with a mace would you? In this case, the Roman warrior had two weapons, his primary being the gladius sword and his secondary being a dagger-like weapon called a pugio. We've used simple box modelling in 3ds Max to create these weapons; using image planes to get the proportions right.

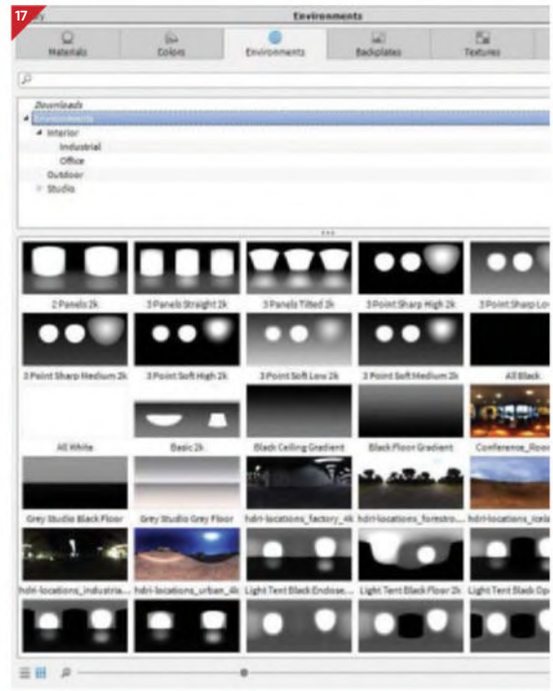
14 Detail the weapons After the base models are done, send them to ZBrush with GoZ. We want the sword to look used so add some chips and scratches with the ClayTubes and Dam_Standard brush. Add some ornate designs using a Standard brush with Alphas and Drag mode applied. The leather part of the sheath was created with Spray mode and Alpha 58.

15 Apply Materials We usually keep it pretty simple when it comes to ZBrush Materials. For this piece we've used only a handful when it comes to ZBrush Materials. The skin and leather pieces use a simple Blinn Material, the armour is Metal 01, and the Cloth is BasicMaterial. Most of the material definition is achieved from tweaking the specularity and roughness settings for each individual piece after we've exported them to KeyShot.



16 Render with KeyShot For rendering, we've decided to use the new KeyShot to Bridge feature in 4R7, which exports the model directly to KeyShot. Go to Render>External Renderer, and click KeyShot. Now, when you hit the BPR button, it will automatically open KeyShot and send over all visible subtools along with their respective materials. From here, we can make adjustments to the material settings.

17 Choose lighting setup KeyShot has a variety of IBL (Image Based Lighting) environments to choose from. You can find them in the Environments tab and you can switch between them by double-clicking the thumbnail image. Adjust the lighting effects in the Settings panel to get the results that you want. To render a final image, hit Cmd/Ctrl+P or click the Render tab at the top. Here, you will be given some options before rendering the image. Just hit Render whenever you're ready! Congrats, you're done!



KeyShot materials

If you aren't happy with the materials that KeyShot assigns your subtools, feel free to add another from the huge library of materials built into KeyShot. Just click and drag them over the object that you want to assign it to. Hold Alt while doing this to preserve your textures.

Texture a realistic superbike and racer

Learn to use Maya and Photoshop to create convincing textures and decals for a motorcycle and its rider

Over the next few pages, we'll go over some fast, effective and fun techniques for creating high-detail textures. We'll be creating an image of a superbike and racer in a full racing suit tearing around a corner. The rider's suit has a complex design with a variety of fabric types, and is covered in logos. The superbike is also covered in decals and composed of different metals and polymers. We'll be using a variety of cool techniques to create textures for everything from the fabric detail, to the logos and decals, to the dirt, rust and debris. We'll start by looking into different theories and tools for quick and effective UV layout. We'll be creating logos and decals then using layout maps to help assemble them into clean and accurate Texture maps. We'll also be creating textures to define our fabrics and to improve our materials. Over the course of this tutorial, we'll cover some fun painting techniques to help get consistently great-looking results quickly. Last of all, we'll use layered textures and shaders to help stack our logos and decals onto our fabric and metal materials.

Our goal here is to play around with various handy techniques, and of course, please feel free to experiment and expand to find what works best for you!


88
MOTOR-BOT
Tocayo
Kunpa
D A
APEX
SIDI

ALDO VICENTE
Accelerator, 2014

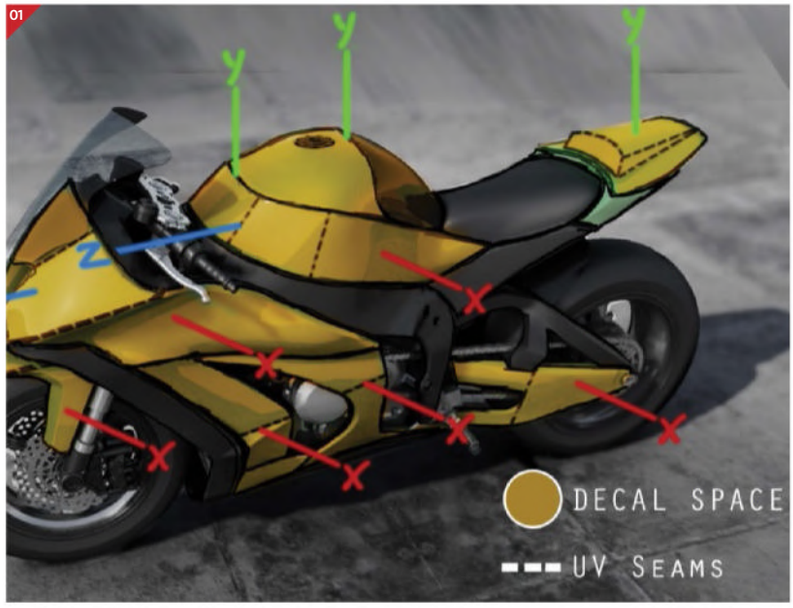
Software
Maya, ZBrush, Photoshop, RoadKill, XNormal

Learn how to
- Lay out UVs
- Paint textures effectively
- Use layered materials and textures
- Use textures to improve your materials

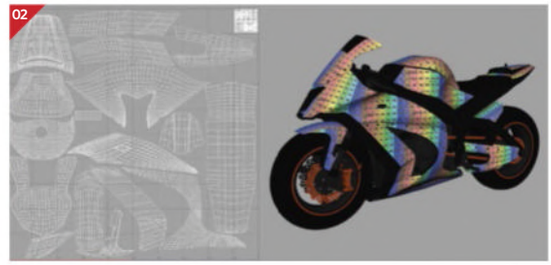
Concept
By working on this project, I hoped to create an image that conveyed a sense of incredible speed, sound and gravitational force. This is hopefully shown with this racer accelerating out of a sharp turn.

YOUR FREE DOWNLOADS from **filesilo.co.uk/bks-798**
- Tutorial screenshots

01 Plan out the bike's UVs A clean, distortion-free canvas for our textures will save us a lot of time and headaches in the long run. Laying out UVs can be a fun challenge when approached with the right kind of strategy. Let's consider our spatial priorities. Each section of our superbike has a solid colour base of either black or green. Solid colour sections need very little map space, meaning that we can designate lots of map space to our decals and consolidate the remaining UVs into a tiny coloured corner of our map. Plus, we can save a lot of time by only refining the UVs on sections that actually feature decals.

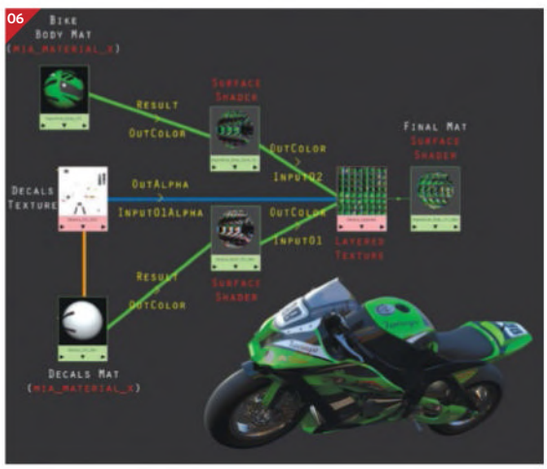


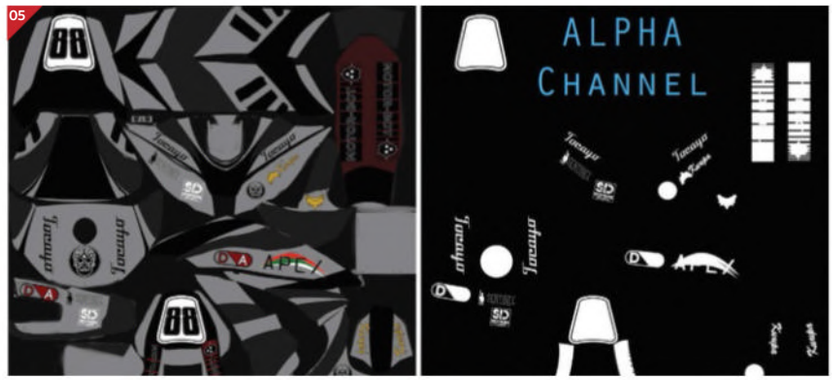
02 Lay out the superbike's UVs In Maya we can use Auto UV Mapping as a starting point. Next we'll select the decal-heavy sections, such as the windshield, the tail and the exhaust panelling, using Planar Mapping to cut them into separate shells. Consider which axis each surface is facing. Let's shrink the solid colour shells into a .1 x .1 corner, keeping the green and black shells separate. Refining our slightly curved decal shells is relatively easy with Unfold and Relax tools. Finally, we can organise our shells for maximum coverage in our 0-1 space. Remember that a uniform scale and orientation will make our lives much easier later on!



Thoughts on UV layout
When it comes to laying out UVs, regardless of what type of project we're working on, it's important to consider things like spacing, orientation, coverage, seam placement, resolution and work time. However, the order of priority is dependent on the individual project. Here, we're creating a pre-rendered still with lots of logos and decals. Therefore, factors like coverage, though important, become secondary as we're free to up the texture resolution. Orientation becomes paramount at this stage as it will save us from painting counter-distortion into our textures.

03 Create a template map Now that our superbike's UVs have been set up, we can start sketching out a template for our decal designs and layouts. Personally I like to use ZBrush's polypaint to sketch directly on the model. Let's bring our superbike geometry into ZBrush, activate polypaint, and coat it in white. Now we can use a Pen brush set to black to doodle in ideas for decals. Once we've found some cool designs and patterns, clean up the sketch and save it out as a map. This map will act as our template, and it will be extraordinarily valuable a little later on.



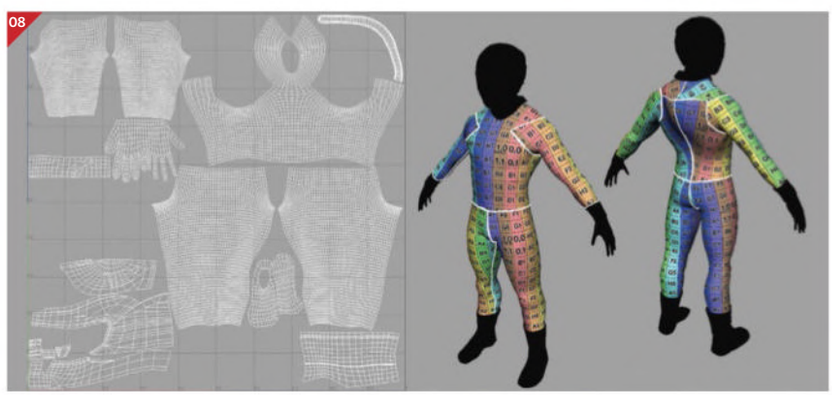


04 Create the decals Here we'll be taking our ideas and fleshing each one out into a polished-looking logo or decal design. A lot of these decals will be used on both the bike and the rider, so we should create and save each one out individually for assembling later on. Logo and decal design is an artform in itself, so remember to gather lots of reference images and inspiration. Let's start by doubling our template's resolution. This will help to ensure that all our logo files are substantially sized. Remember to keep each design on a transparent background.

05 Assemble the decal texture Now that all our decals have been fleshed out, let's bring them together to assemble our texture in Photoshop. Open up our standard-resolution template and start importing decals. Once we've scaled and sorted them into place, remove the template and merge all the decal layers into a single Diffuse texture. Create a new Alpha channel, then select the positive space of our Diffuse texture and fill it white in the Alpha channel. Save the texture in an image format that supports Alpha channels, like 32-bit TARGA.

06 Add decal textures to our materials In Maya, create a decal material and a layered texture shader, then drop them into the Hypershade workspace with our superbike's body material. Next we'll bring our decal texture in with a 2D File node and plug it into the Diffuse Color of our decal material. Now we can stack the two materials into the layered texture shader, with the decals on top (left). Next, connect the Alpha out from our decal texture to the layered texture shader's Input01>Alpha. Last of all, we'll create a surface shader and plug the layered texture into the Outcolor and apply that to our superbike.

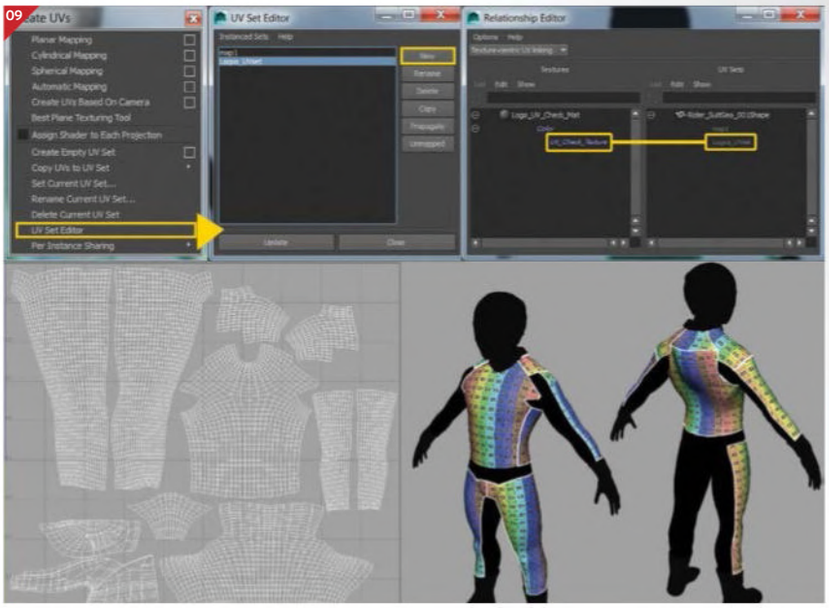
07 Plan the racer UVs Creating textures for the racer is going to be more challenging than the bike because in addition to a decals/logos texture, the racer needs textures for suit and fabric details. For this, we'll need to create two UV sets for our racer geometry. The logos' UV set will be similar to the one we created for the superbike. Maximise space for the areas with logos, and confine the empty sections to a small corner. The second UV set will be a more traditional UV layout, with a complete, even unwrapping. This will be our suit UV set.



08 Create the suit UV set Let's create a new UV set for our suit in the UV Set Editor window. We want the suit UV set to be a full, traditional unwrapping of our geometry. This UV set will be used to create textures for the suit details and fabric. We can break our model up into a torso and four separate limbs and then Unwrap and Relax each island like we would a cylinder. I like to keep a vertical orientation on everything, as this will make it easy to keep a consistent fabric pattern while creating our textures later on.

09 Create the logo UV set The next step is to make a new UV set for our rider geometry in the UV Set Editor. Let's select our suit geometry and run an Auto Unwrap. Now, let's use Planar mapping to lay out the planes that will feature the majority of our logos. We'll take the remaining UVs and shrink them into a small corner. Let's make sure we have uniform scale and orientation on our logo UVs. Last of all, we'll flatten out our logo canvas. Maya's UV Texture Editor can get the job done, but remember to try out some of the other great UV tools outside of Maya.

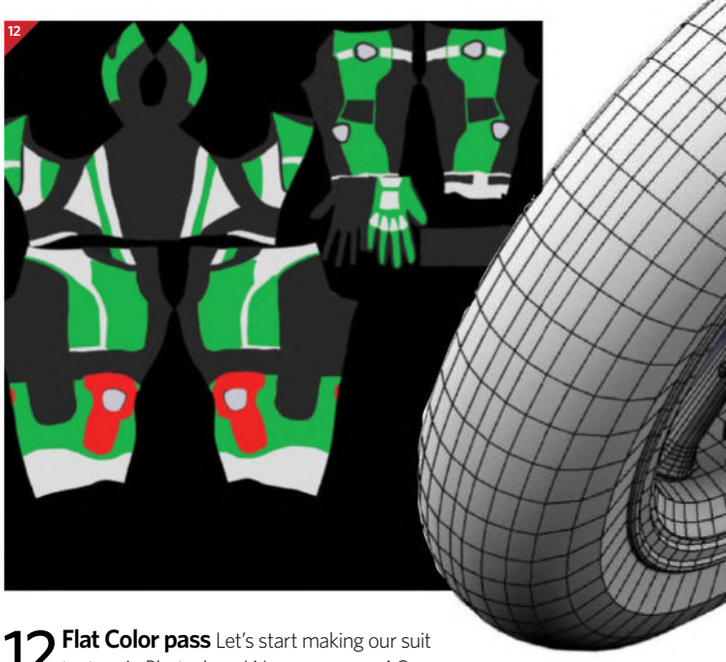
10 Normal maps, Displacement maps We'll be creating our Displacement maps, Normal maps and AO maps in ZBrush. These maps will be our guides for painting our suit textures. Let's import our suit geometry, making sure the suit UV set is assigned. Now we can sculpt in our suit detail to project onto our Normal and Displacement textures. At this stage, we don't want fabric texture. We only want to convey the way the suit is stitched and constructed, and how the fabric folds and bends around the stitching. Once our details are ready, generate Normal, Displacement and AO maps.



Manage your colours

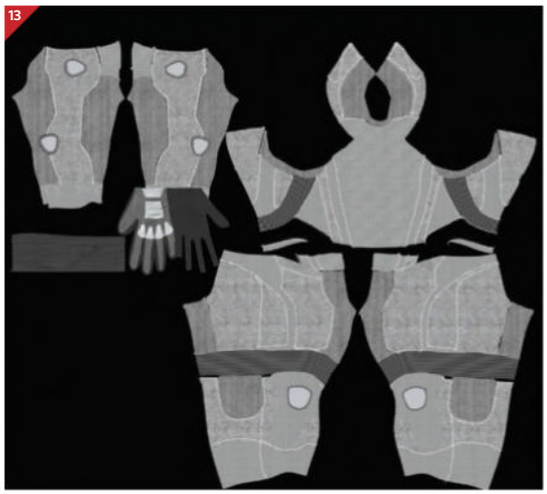
I used a linear workflow to ensure that all the incoming and outgoing colours – whether they're from textures or colour swatches – are balanced into a uniform, linear colour space. Understanding colour management and how to establish a linear workflow is important for getting the most realistic and accurate results, so it's absolutely worth researching. For a quick setup, let's enable Color Management in the Render Settings window. Now we can use Gamma Correct nodes to balance our colour swatches. We can label our colour textures as sRGB, and we can label our Bump, Normal and Displacement as Linear sRGB.

11 Sketch the logo template Similar to how we sketched on the bike earlier on, we want to sketch in a texture to use as a guide for our logo placement later on. In ZBrush, be sure to import a suit model with the logo UV set. Again, we'll coat the mesh in white and draw with a Pen brush set to black. This time around, we need to consider the size and aspect ratios of our existing logos and find cool ways to distribute them across the surface of our suit. Note that unlike the superbike decals, these logos are asymmetrical, with each appearing on one side only.

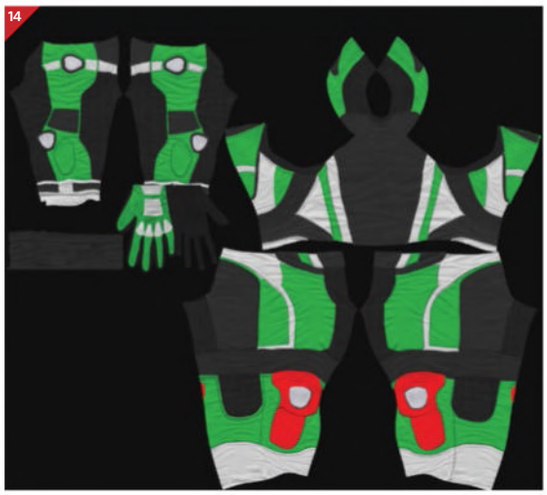


12 Flat Color pass Let's start making our suit texture in Photoshop. We can use our AO maps as a template while painting in our textures. Start off with a flat Color pass. We are essentially filling in the base colours of our fabric panels. In this step we're not concerning ourselves with details like wrinkle shading, fabric type, grunge, seams or stitches. We should make a selection out of our AO map and simply fill in with flat, solid colours. Be sure to keep each different colour on its own layer for further adjustment later on.

13



14



15 Create the logo texture With our suit textures ready to go, we can move on to creating our logo texture. This process, again, will be similar to creating the decal texture in Step 5. Let's load our logo template texture into Photoshop and start importing our pre-made logos. Use the logo template that we made earlier in order to quickly pop the various logos into place. When all logos are positioned, we can hide our template, merge the layers and create a new Alpha channel using the positive space from our logos layer. Before moving onto the next step, we can save out our logo texture as a 32-bit TARGA.

Bump map tips

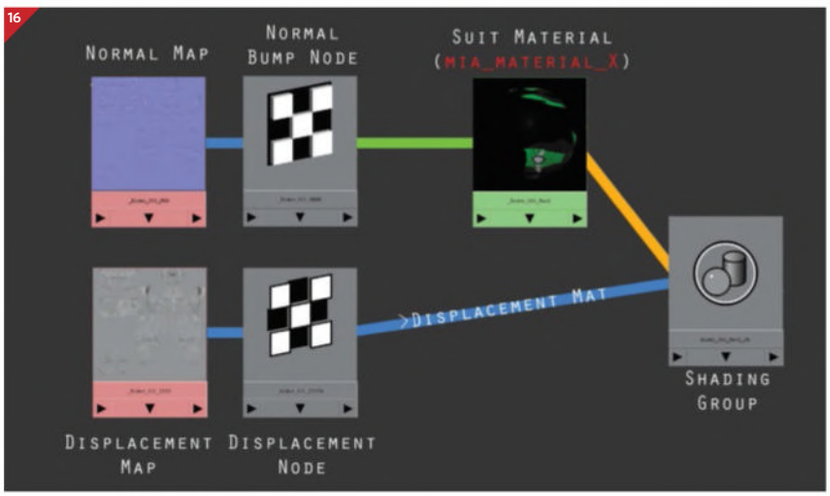
There are some great tools out there for creating effective Bump maps from photo images, and it can be fun and helpful to go out and photograph new textures. Having control over framing, scaling and lighting can be very beneficial when you're working on such a highly detailed image. We can manipulate our levels in Photoshop to create some really effective Bump maps. For our fabric Bump map, I collected photo textures for each of the fabric types in our suit, then balanced out their values, adjusted scaling for accurate comparative proportions, and then set a uniform orientation. This provides us with an excellent starting point for creating our Bump maps.

13 Fabric Bump pass Let's create Bump textures that will convey what type of fabrics are in our suit. We're keeping this Bump detail separate from the suit structure detail in the Normal map. To create our Bump maps, we can use photo textures of the actual fabrics that we're trying to simulate. There are some great tools out there for creating Bump, Normal, or even Displacement maps from flat images, but I've also found it easy to create excellent Bump maps by simply desaturating photos and manipulating the levels. We should balance the levels for all our fabrics to fit within the same range.

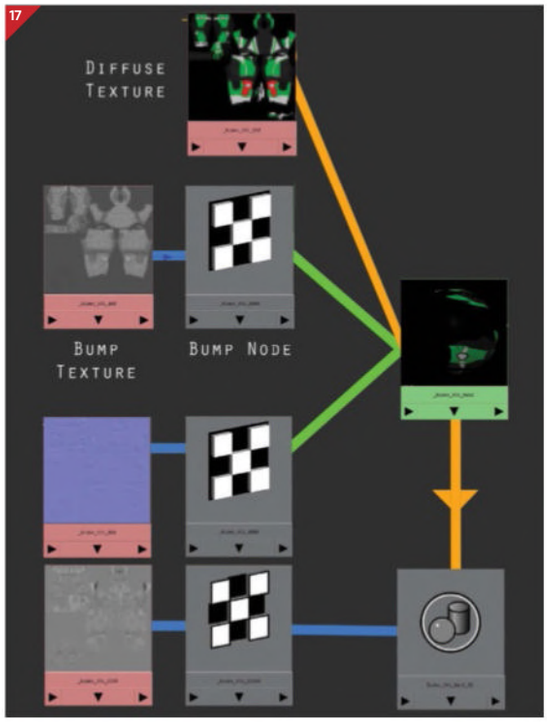
14 Fabric Color pass I've found that we can make our Bump textures even more convincing by integrating the same detail into our Diffuse texture. First off, make a copy of our fabric Bump pass and set them over our solid Color pass. We can set our fabric colour layer's blending mode to Multiply to bring the fabric patterns into our colours. Next, we need to adjust the value levels of our fabric Color pass. When set to Multiply blending, adjusting the levels affects the Opacity. The more white, the more subtle our fabric colour texture will be.

15

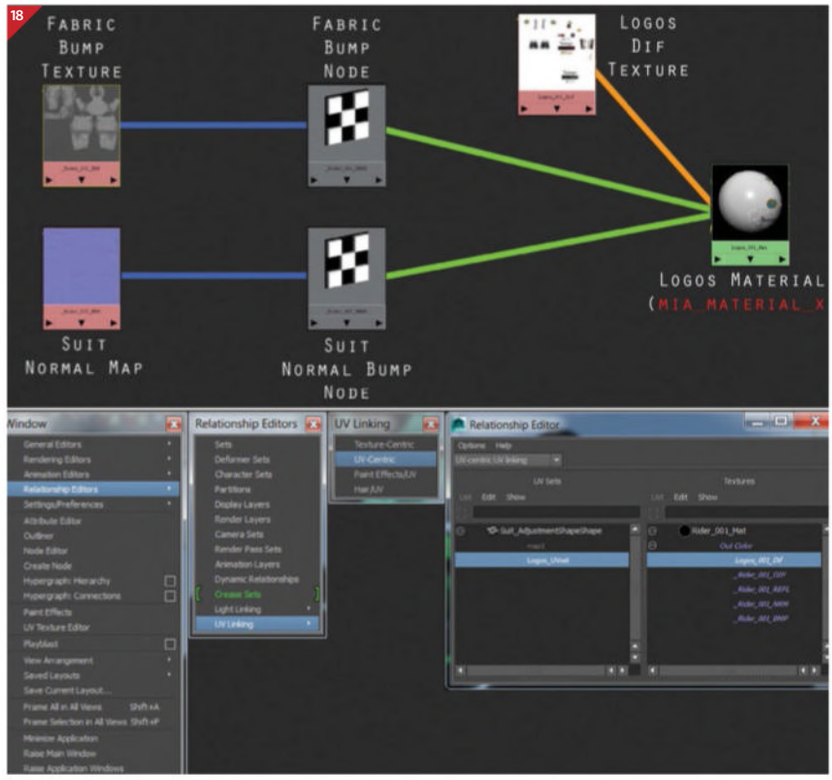
16



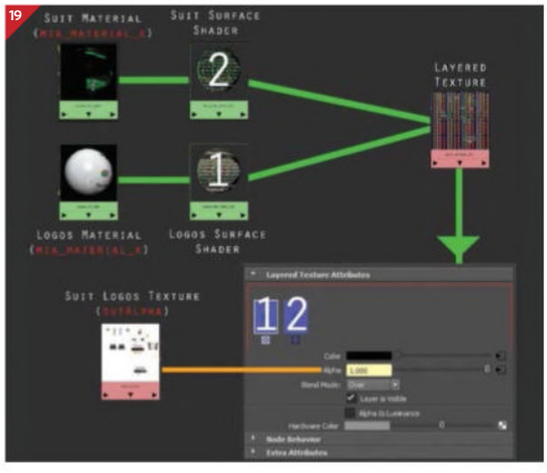
16 Prepare the suit material It's now time to create the base material for our suit. I used mia_material_x. We'll be adjusting the Shading attributes later on. Create a 2D Bump node and plug in our Normal map, setting the Bump type to Tangent Space Normals. Let's apply this material to our suit geometry and then apply a Displacement node to the Shading group. Plug our Displacement map into the Displacement node. Remember to label Displacement and Normal maps as linear maps, and to designate them the suit UV set. Lastly, we can set up Subdivision Approximation on our geometry to refine the effects of our Displacement map.



17 Apply the suit textures Now that we have our base suit material set, create a 2D File node and bring in our suit's Diffuse texture. We'll be plugging our Diffuse texture into our suit material Diffuse attribute. Be sure that the Texture Input Profile is set to sRGB. Next, we'll create another 2D File node and bring in our suit Bump texture. Our Bump texture should have a Linear Input Profile. Since this is very subtle detail, our Bump value should be set low. Finally, connect the 2D Bump node into the second Bump slot on our suit material.



18 Create the logo material Make a material for the logos. Looking at reference, we can see the logos' material sits over the suit fabric, so this material will not share the suit's fabric detail, but it will carry the suit's structural detail. So, let's start by applying our Normal map. Next, we can us a 2D File node to bring in our Logos texture and plug it into the Diffuse slot of our Attribute Editor. Our logos' texture will not be using the suit UV set, so let's be sure to assign the logos UV set in the UV Texture Editor.



19 Layer our materials With our suit and logo materials ready, it's time to start layering. Drop the two materials into the Hypershade workspace with a new layered texture shader. Before we drop them into our layered texture, we need to plug the result of each mia material into the Outcolor of a new surface shader. Now we can drop these surface shaders into the layered texture, with the logo material on the left-hand side. Finally, let's select the logos' texture file, and connect the outAlpha, to the Alpha attribute of the logos' surface shader (input01) in the layered texture shader.

20 Final suit material Now we'll set up our final rider material in a new surface shader. We'll take our layered texture shader and plug it into the Outcolor of our final surface shader. Now we'll apply our final surface shader to our suit geometry. At this point, we have our suit's fabric and structural detail in place, with the logos ironed over the top. Since our geometry already has subdivision approximation set up, the last thing we need to do is to connect the Displacement node we set up earlier and plug it in to our final materials Shading Group.

Adding layers

In these examples we only stack two materials at a time, but if we want to add detail, we can use the same process to continue stacking new materials on top of what we already have. If we want to add grunge with its own material properties, start by creating a grunge texture with an Alpha channel, featuring only the grunge detail. In Maya we can create our new grunge material, plug in our texture, and then stack it on top of our layered texture. Finally, we can plug the texture outAlpha into the layered texture.

> You can't always get the result you want when you're only using procedural textures, but using them as much as possible can let you stay flexible

Maya, V-Ray

Tashina van Zwam
framecandy.nl

As a 3D artist for MediaMonks, Tashina's work mostly focuses on shading, lighting and texturing

YOUR FREE DOWNLOADS
from **filesilo.co.uk/bks-798**

- Tutorial screenshots
- Scene file

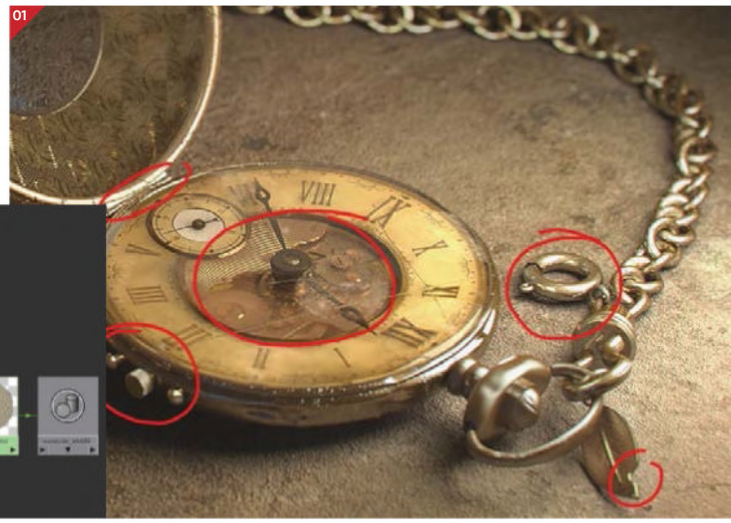
Texture an antique pocket watch

It's often the little details that really make an image shine and become something truly unique. It could be small imperfections that give that last little touch or that shimmer of light that portrays the right mood. There are many things that you need to consider when you are creating a realistic image, but it's often those finishing touches that are hard to get right and really master. In this tutorial we'll point out some tips and tricks that can help you achieve a better critical view in your artwork.

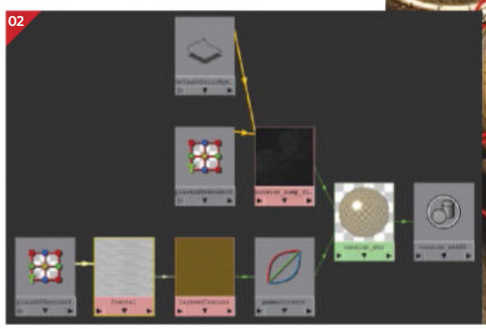
We'll address how small details or a shader can direct the eye of your audience. We take you through why it's all about finding the right balance in your image. Furthermore we will give an insight about how and when to add elements or sometimes remove things to get a better end result. In that process we will take you through how we work with procedural textures to help tweak that balance in a more efficient and controllable way. And finally we will take the lessons learned into the world of compositing to see how they translate and help you with the final render.

Try to apply this to your own work and see the difference it will make. You can make a big change by taking the time to pay attention to the extra details.

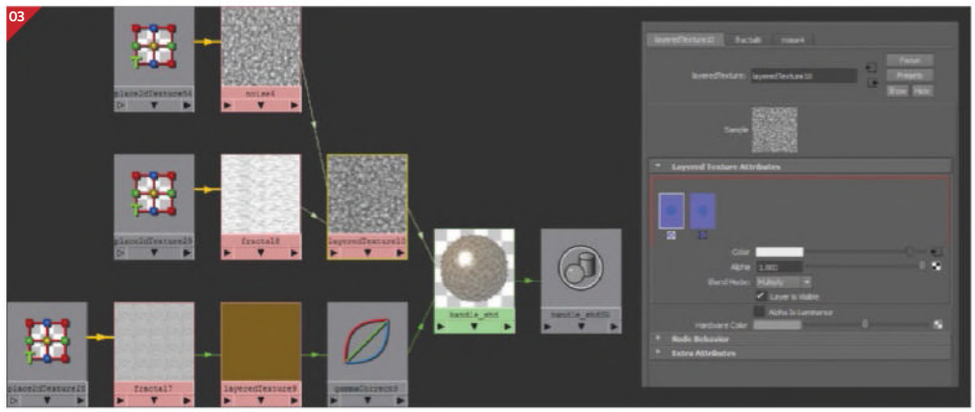
01 Details that actually matter Not only will the shading and lighting give you a realistic feeling, but so will the amount of objects and details in the scene. One of the goals for this render was to make it interesting enough so that's why small details were added. For example, look at the chain and also the wheelwork inside the watch. It's different for every scene, but if you don't add enough details you soon get the feeling that it's not complete and with that you lose the ability to create a believable image. Balance your image well to ensure you keep your audience's interest.



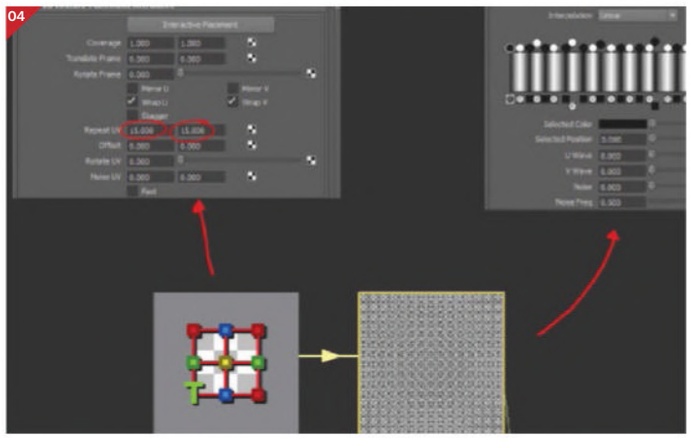
02 Gold shader This pocket watch needed a beautiful gold shader but taking it to a higher level required some extras which could be achieved by adding different kinds of Bump maps and other textures. One way of losing believability is when you use repeating textures which you do not often see in the real world. A lot of times you can get a nice Bump map by using things like a fractal or noise. Tweak the settings and you can get some interesting results. Of course you can also use images from outside the program or do a combination of the two.



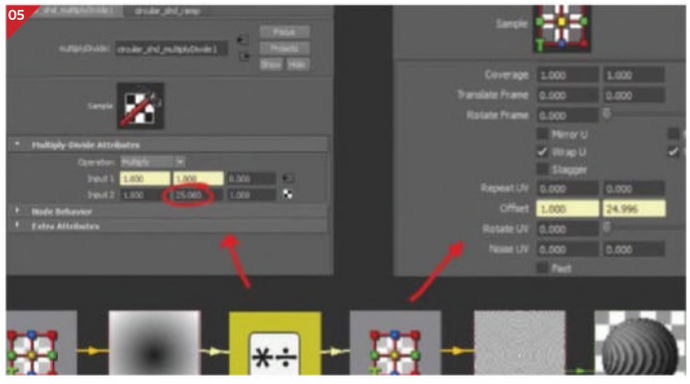
03 Tweak with procedural textures It's great working with procedural textures because of their flexibility. They are solely built based on a set of nodes in the 3D application. You can't always get the result you want when you're only using procedural textures, but using them as much as possible can let you stay flexible and perform quick tweaks. Another thing you can do is combine 2D textures, like fractals and noise, and use them in a layered texture. In this way you can multiply, add or subtract layers. This can give you the possibility to create some fantastic procedural textures by only using everything that the 3D application offers you.



04 Easy adjustable circular shader To divide the details in the image nicely we'll add some details to the top cover of the pocket watch. We'll make a circular pattern to use it as a bump of the shader. A simple way to do this is using a circular ramp with a black-and-white striped pattern and create the repetitions manually. The downside to this is that it will become time-consuming if you want to change the amount of repetitions, especially when working with higher numbers. It's better to build a procedural network that allows for changing the circular repetitions easily.



05 Create the repeatable circular shader The reason to use procedural shaders is for flexibility. Let's make it easy to change the amount of circular repetitions by using a trick to remap the UVs in our procedural network with a place2DTexture node. First we disable its default offsets by setting repeatU and repeatV to zero. Then we define our own offsets as inputs to offsetU and offsetV. For this, we use a simple black-and-white circular ramp. Add in a multiplyDivide node to easily control the amount of repetitions. This way you will have more flexibility and save time.



Rule of Thirds

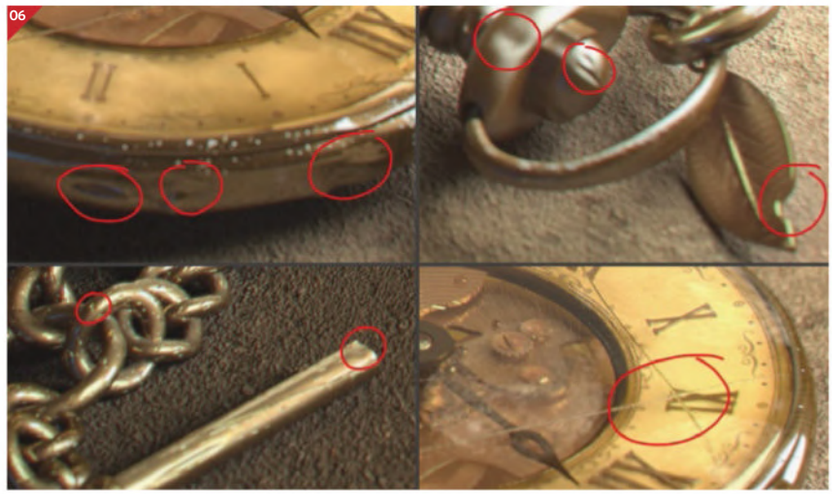
The guideline of Rule of Thirds is great for balancing the composition of your image. The guideline suggests to divide your image into nine equal parts, with two horizontal and two vertical lines, and that important elements should be placed on the intersections of those lines. It's believed that aligning objects to these points balances your image and will enable a viewer to interact with it more naturally. It's more of a rule of thumb, so don't force yourself into unnatural compositions trying to use this rule, but use it when it feels right.

06 Tricks to make it realistic One of the most important things in making a realistic image is to create imperfections. Add different kinds of little dents and bumps but don't overdo it. We only need to feel a sense of it, but it should not take all our attention. Most objects in the real world are not perfect so when you create an object try to make little adjustments to everything. Let's say you want to create a wooden beam, don't just make a rectangle and give it a texture – give it all sorts of imperfections. Add enough edges and move some vertices.

07 Render passes For control during compositing you definitely need render passes. By rendering your image with its separate passes like diffuse, lighting, GI and reflection you can re-create your exact beauty layer. With these rendered separately you can still control how these layers are combined during compositing. For example tweak the amount of shadow on the ground or the diffuse colour of a single object (using a multimatte) and apply the correct lighting on top of that. Get as much information out of your 3D application as you require to have all the control you need.

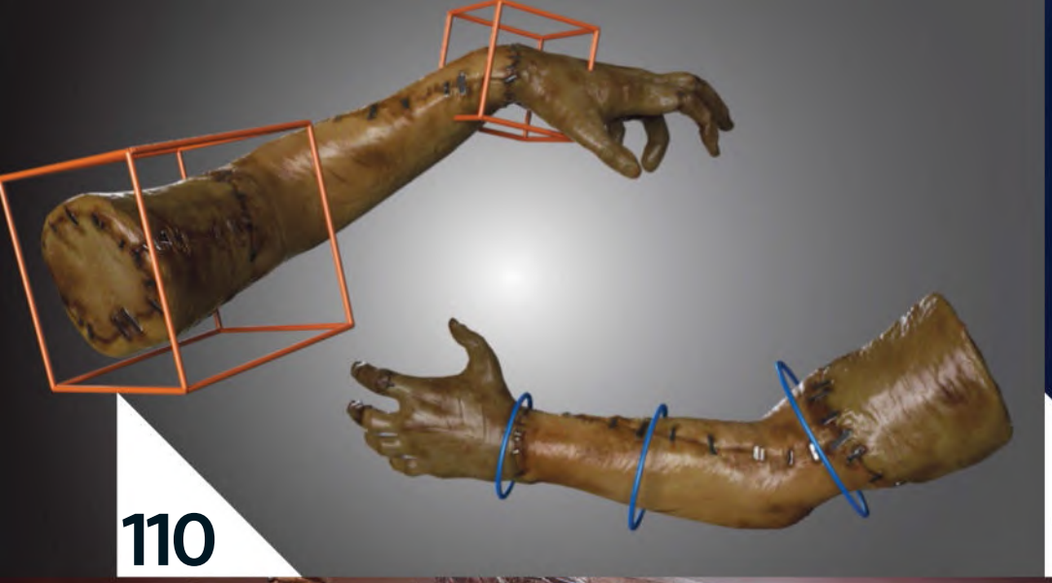
08 Chromatic Aberration In real life when you take a picture, the lens of your camera can give distortion in which there is a failure to focus all colours to the same point on the camera's sensor. As colours have their individual wavelengths, they refract differently through the lens. By using a zoom blur on the red channel you can offset this colour with the longer wavelength towards the edges of the screen and fake the lens refraction. You don't want to exaggerate it too much but if it's done right it can really boost the believability of your image.

09 Dust and particles What's also interesting is the use of dust and particles. Normally objects are never totally clean, but will always have some dust on it and this also applies to the lens of your camera. By adding dirt you will make your image more realistic because now it's not perfect anymore. This also applies to particles floating in the air. In real life when there is a sunbeam, you will see quite clearly that there are little particles all over the place. In this case there are no sunbeams but there is light so there will always be some dusty particles floating around.



Weighing your image

Another way to balance your image is looking at the visual weight of its elements. When an object is positioned to the right of your image in an empty space the weight shifts to the right. To balance it out you'll have to add weight to the other side. In this case you have the detailed inside of the pocket watch on the right and the top cover on the left. To add enough weight to the top cover, try adding a detailed pattern. Similar weight balancing is going on with the chain in the bottom left and top right.

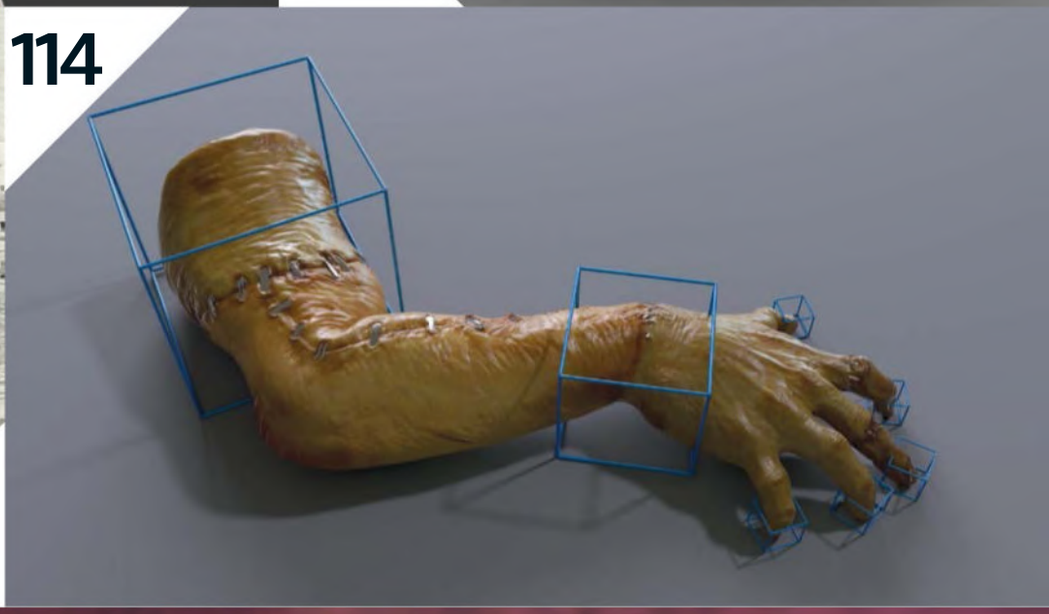


Rigging & Animation


104


100


118


114

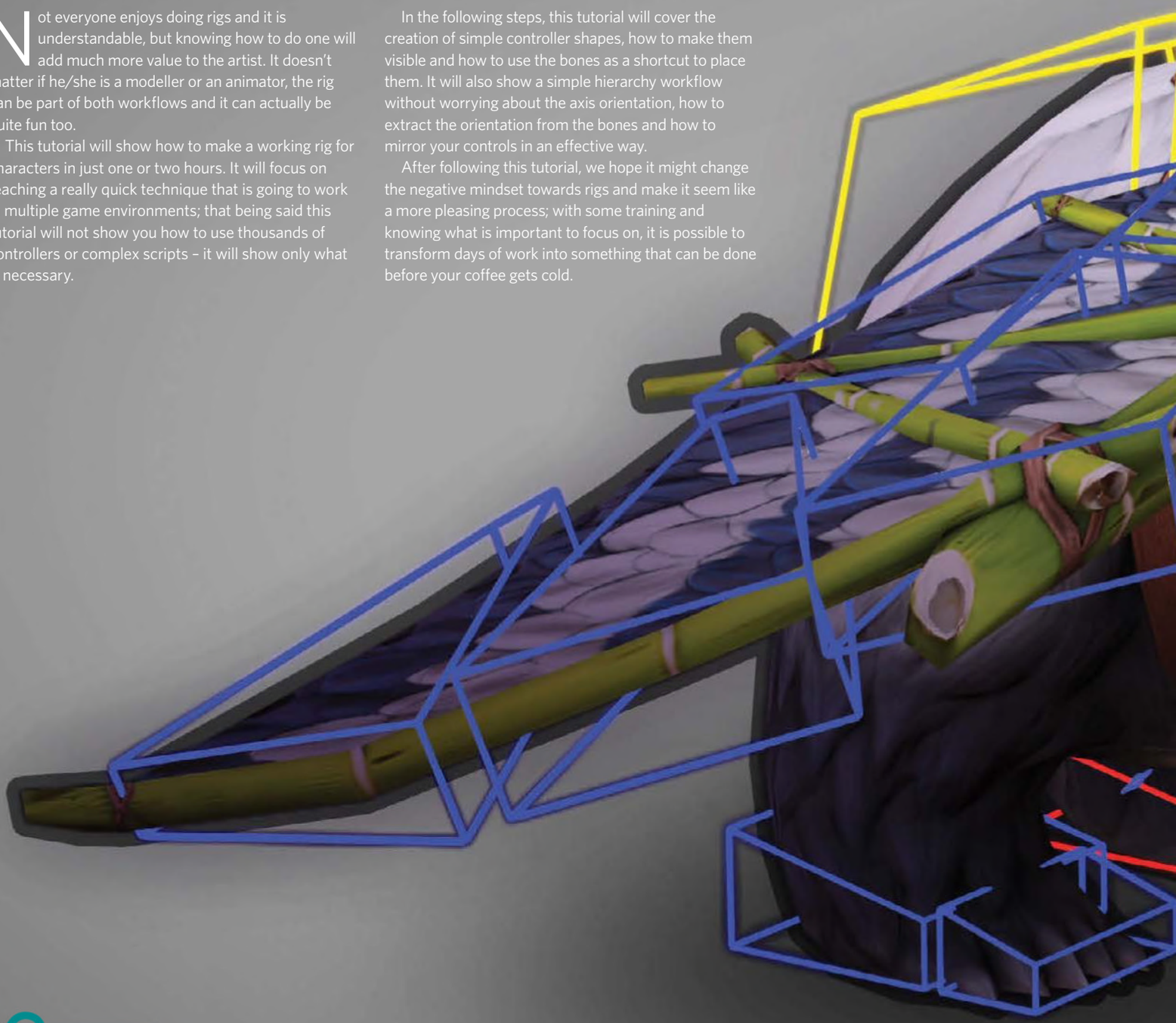
Build a quick rig for videogame characters

Not everyone enjoys doing rigs and it is understandable, but knowing how to do one will add much more value to the artist. It doesn't matter if he/she is a modeller or an animator, the rig can be part of both workflows and it can actually be quite fun too.

This tutorial will show how to make a working rig for characters in just one or two hours. It will focus on teaching a really quick technique that is going to work in multiple game environments; that being said this tutorial will not show you how to use thousands of controllers or complex scripts – it will show only what is necessary.

In the following steps, this tutorial will cover the creation of simple controller shapes, how to make them visible and how to use the bones as a shortcut to place them. It will also show a simple hierarchy workflow without worrying about the axis orientation, how to extract the orientation from the bones and how to mirror your controls in an effective way.

After following this tutorial, we hope it might change the negative mindset towards rigs and make it seem like a more pleasing process; with some training and knowing what is important to focus on, it is possible to transform days of work into something that can be done before your coffee gets cold.



YOUR FREE DOWNLOADS
from **filesilo.co.uk/bks-798**
- Tutorial screenshots
- Video tutorial

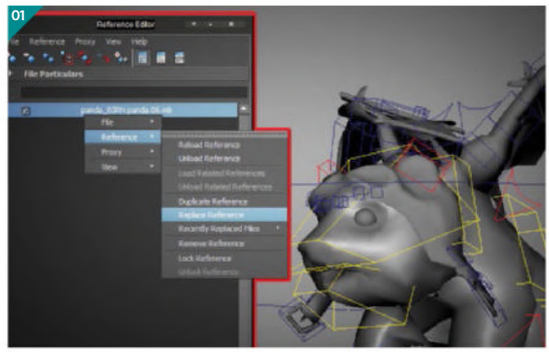
Maya

Thiago Vidotto
tvidotto.com

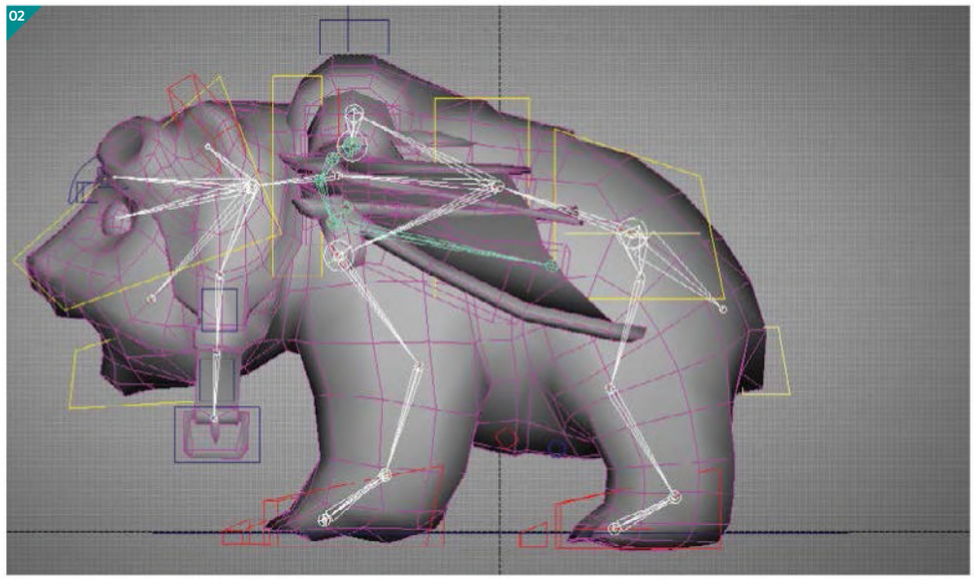
Thiago Vidotto is a Brazilian self-taught generalist working as a freelancer for videogame art

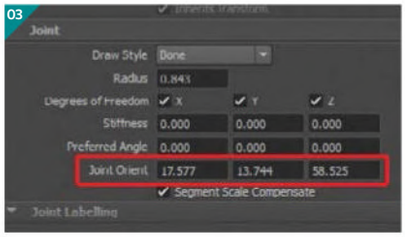
01 Keeping it editable Keep the rigs as simple and editable as possible. We prefer spending more time on the animation, and because of this we can get back to the rig and change it. The first important thing to make this happen is to never animate directly on your rig file; always reference the rig inside the animation scene. This is so that every time we change the rig we only need to update the reference. Working like that enables the animation to be produced in parallel with the rig after an initial stage.

02 Bone placement It is important how you place the bones as it can affect the entire rig and make animation process a nightmare. A good rule is to follow the anatomy of a living animal that looks like your character. Of course most of the time it will need some interpretation, but referring to a real animal will make the final result more believable. Sometimes we need to be economic with the number of bones, so it is good to avoid finger bones or complicated wing structures.

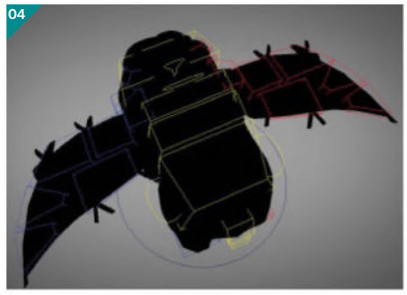


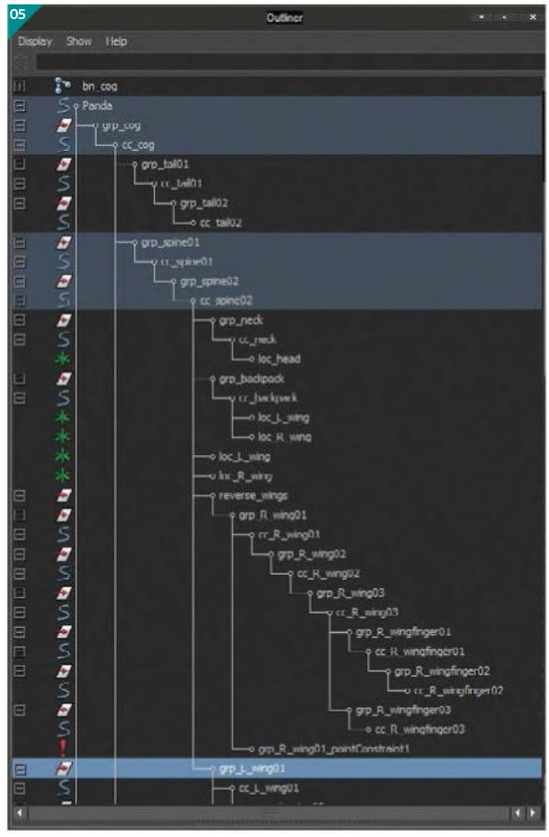
03 Bone orientation The orientation of your bones is also important for an easier animation process. Most of the main body bones usually follow one of the orthographic views, but some will require a special rotation angle – on this rig this was needed for the eyebrows and ears. To rotate them, make sure that you don't use the rotation tools; your bones should have zero rotation. They need to be oriented using the Joint Orient fields inside Joint Parameters in the Attribute Editor. Select the field you want to rotate and with the Cmd/Ctrl key pressed, use the middle-mouse button to modify the values.





04 Controllers Keep the controllers simple. The focus here is to be quick and to not waste time on fancy shapes – the rig needs to be easy to understand and easily manipulated. Work mostly with cubes made from curves and circles from the NURBS menu as they can be scaled and slightly changed to fit the shape of the body. You can use red and blue to identify the right and left sides of the controllers, yellow for the middle and a less saturated colour for secondary controllers.





How to make cube-shaped controllers

First, create a polygon cube, then open your script editor window and clean it. After that go to Create>CV Curve Tool and click on the option box. Then in Curve Degree select 1 Linear. Now, with the V key selected, click on each vertex of the cube. Don't worry about overlapping lines. When you're done, press Enter and your cube controller will be ready. To avoid having to make a new one every time, select the line created in the script editor and with the middle-mouse button drag it to the shelf, selecting MEL as the script type.

05 Hierarchy of the controllers After creating all the shapes for the controllers, it is necessary to organise them in a hierarchy of movement. Usually this follows the bones' hierarchy and it can be done in multiple ways. The way we like to do it is to parent the controllers to the respective bone, zero the translations, and with the controller selected, create a group (Cmd/Ctrl+G). After unparenting the group from the bone, we will have the controller zeroed but following exactly our bone's orientation. Now stack groups inside the previous controller making the hierarchy chain alternate into groups and controllers.

06 Break the hierarchy As said before, the controllers usually follow the same hierarchy of the bones but a situation where this may not necessarily be the best option is on the head. When we are rigging the head we want the controller to follow the position of the neck but not the rotation. To do that, we need to place a locator parented to the neck controller and use it to define only the position of the head's group, instead of parenting it under the neck hierarchy. This will enable us to easily control its rotation and where the character is looking.

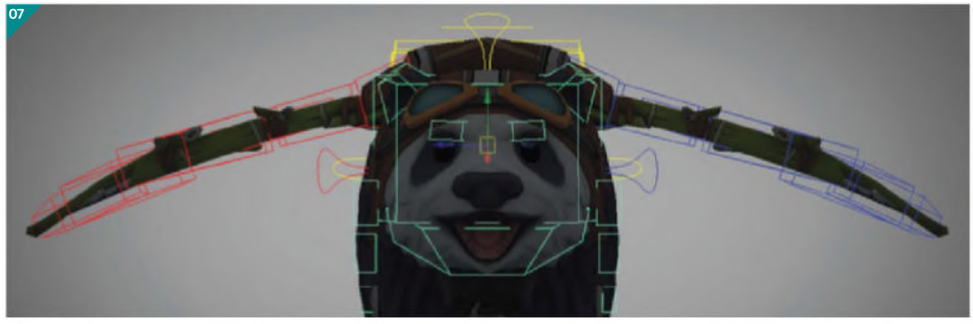
07 Mirroring Mirroring the controllers saves a lot of time but if not done right it may give you some extra problems during the animation process. A good mirrored controller will keep the same rotation and translation behaviour when keyed with the exact same values. However, if you create a controller on the opposite side of a character with the exact same method, you will not be able to copy the animation curves from one side to the other. To fix this you can parent all duplicated chains to an extra group with a negative value on the x scale.

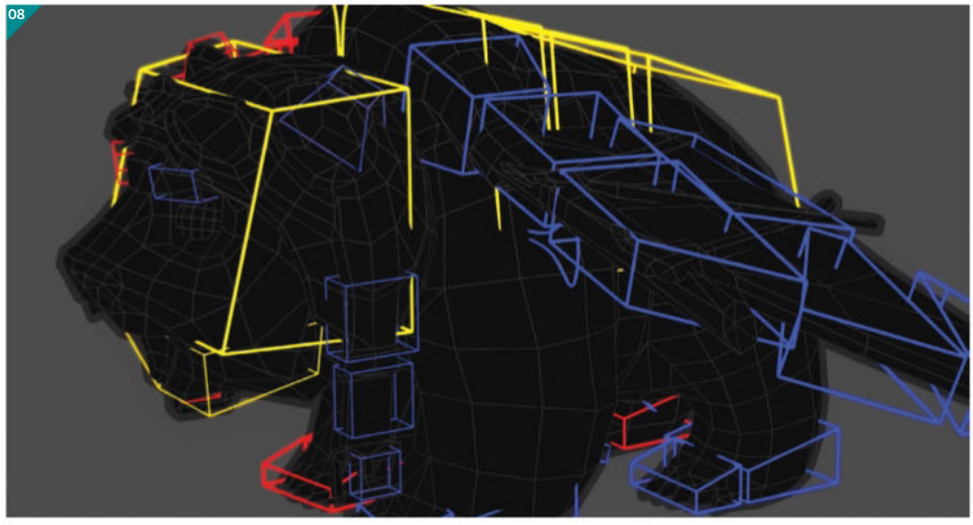
08 Finish the controllers Before constraining the bones to the controllers, it is good to make a test pass on all of them. It's not uncommon, when doing a repetitive process like organising the hierarchy chain, to forget a controller is unparented or that it has some rotation values. All controllers must have zero rotation and translation, but that is not true for the groups: they need to have the info from the bones made in Step 5. After this stage is done, almost all of the bones can be constrained with parent constrain, the ones using a mirrored controller should use a point and orient constrain.

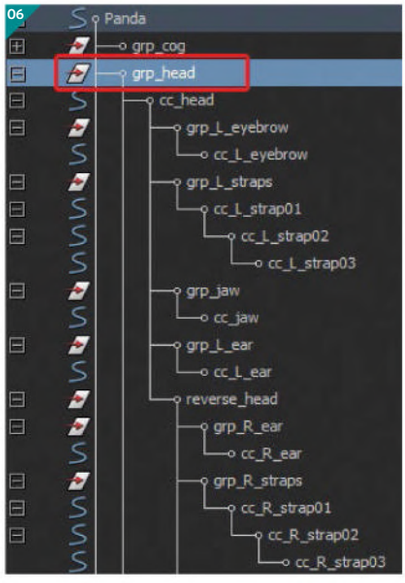
09 Intermediary skin Now do a quick skinning and don't aim to make it perfect before starting the animations. A low-poly model can be hard to skin sometimes, and you will only know if it's working when testing some animations. After doing a smooth bind on the mesh, save several animation poses on the timeline before tweaking the skin. Now start adding weight to the external parts of the rig, like the claws, then lock the influences and keep going higher on the hierarchy chain. It requires a lot of trial and error, so keep testing the poses often.

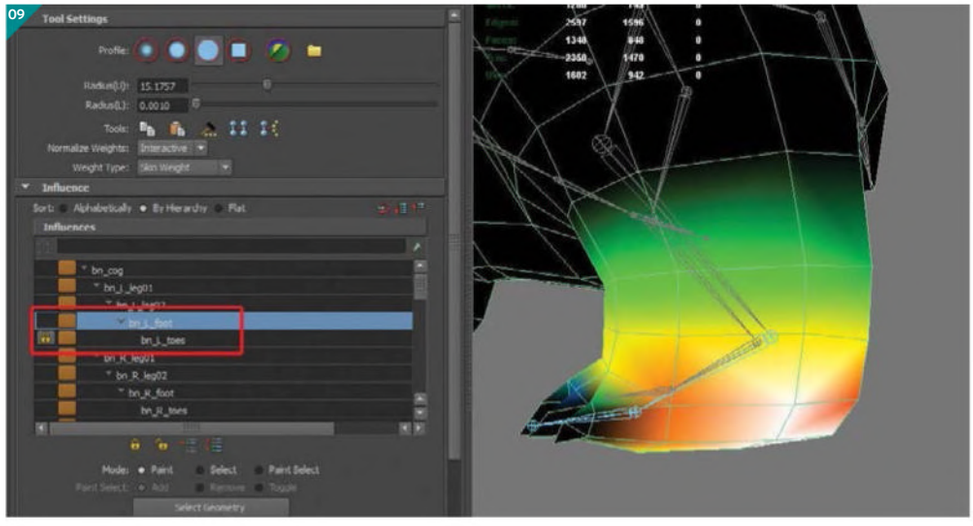
Colouring the controllers

Using colours on the controllers make them more recognisable (usually blue for left side and red for the right), which is really useful when you start animating with the texture. The use of saturated colours keeps their visibility even with the new visual distraction. Maya is not good at making the colouring task easy so the best way is by using a script called dpColorOverride created by Danilo 'Nilouco' Pinheiro (**nilouco.blogspot.ca**). It creates a small window, where you just click once on the colour you want and it will colour your controller.





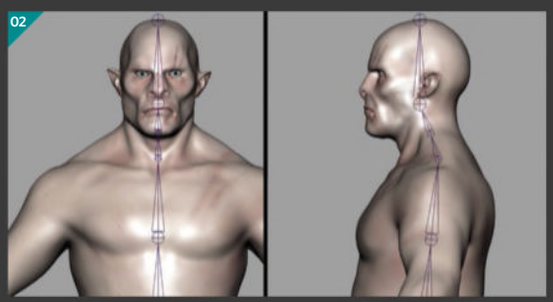




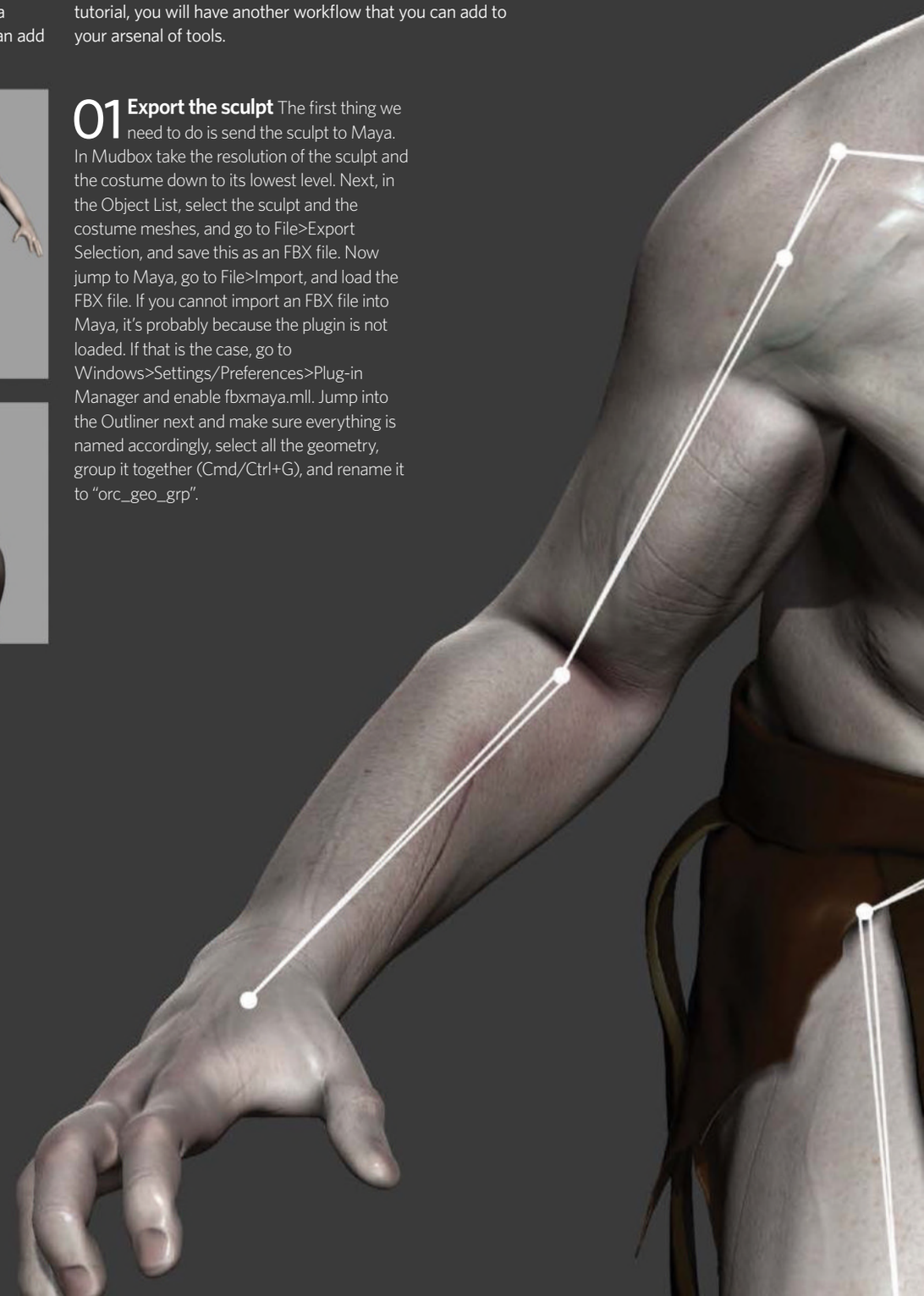
Pose a sculpt using joints

When you are creating a sculpt, building it up in a generic T-pose usually simplifies the sculpting process, and among many things helps to ensure that all of your model's limbs are in proportion. However, this also makes for a stiff-looking character. In this tutorial, we'll look at how you can go about adding a bit more life to a sculpt whilst still being able to go back to the default T-pose, as that is how you would want to pass the model should it need retopologising and rigging for animation.

Now there are many different ways in which to pose a sculpt that you've made. For example, in Mudbox you can add joints and use falloffs; in ZBrush you can use Transpose or a ZSphere rig. The preferred method here, however, is to take the lowest resolution of the sculpt into a 3D package – in this case into Maya – and take advantage of the program's specific rigging tools. Although the set-up time for creating the initial skeleton and then binding the mesh to the skeleton may be slightly longer, once complete, you can create a wider range of poses more quickly than if you were using the default posing tools in Mudbox or ZBrush. Hopefully, by the end of this tutorial, you will have another workflow that you can add to your arsenal of tools.



01 Export the sculpt The first thing we need to do is send the sculpt to Maya. In Mudbox take the resolution of the sculpt and the costume down to its lowest level. Next, in the Object List, select the sculpt and the costume meshes, and go to File>Export Selection, and save this as an FBX file. Now jump to Maya, go to File>Import, and load the FBX file. If you cannot import an FBX file into Maya, it's probably because the plugin is not loaded. If that is the case, go to Windows>Settings/Preferences>Plug-in Manager and enable fbxmaya.mll. Jump into the Outliner next and make sure everything is named accordingly, select all the geometry, group it together (Cmd/Ctrl+G), and rename it to "orc_geo_grp".



Maya, Mudbox

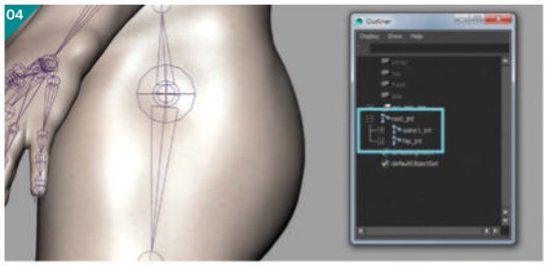
Jahirul Amin
jahirulamin.com

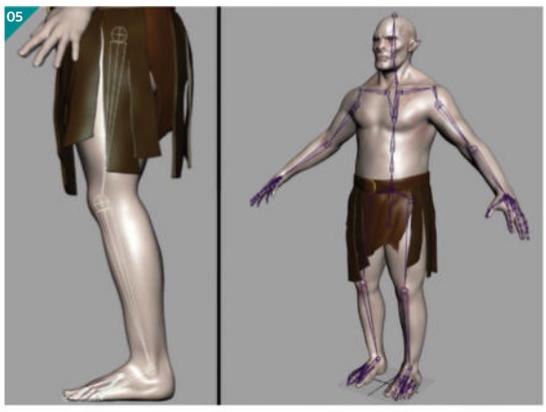
Jahirul Amin is a 3D trainer at Double Negative and has a passion for rigging and animation

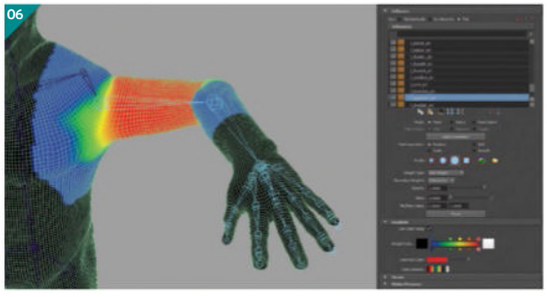
YOUR FREE DOWNLOADS
from **filesilo.co.uk/bks-798**

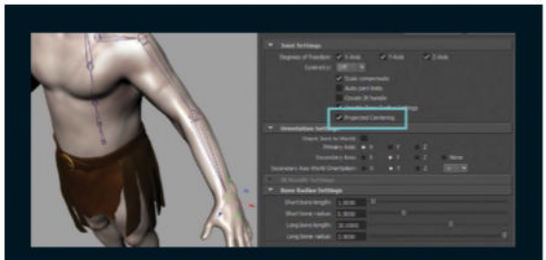
- Tutorial screenshots
- Maya & Mudbox files
- Video tutorial

> You can create a wider range of poses more quickly than if you were using the default posing tools in Mudbox or ZBrush









Projected centring

If you are using Maya 2015, you can take advantage of Projected Centering. This will let you create your joints in the Perspective view and, automatically, the joints will centre themselves in the mesh. To enable this, go to Skeleton>Joint Tool (Options) and enable the checkbox.

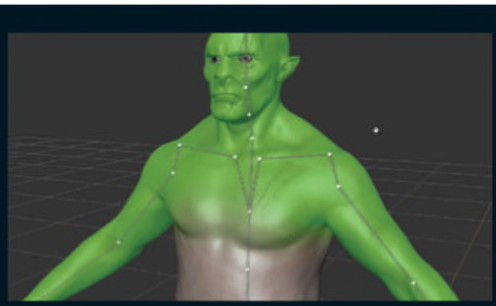
02 Create a joint chain for the spine, neck and head joints Now that we have the sculpt in Maya, let's create the skeleton for the spine, neck and head. Press F2 to switch the toolset to Animation, and then make sure you are working in the Side view. Now go to Animation>Joint Tool, and using the LMB, make a series of clicks from the root of the character (slightly below the belly button) upwards to the top of the head. Click Enter to come out of the tool. If you make a mistake whilst in the tool, use the Backspace on the keyboard to remove the last joint and then the LMB to continue the joint chain. For our joint chain, we've created four joints for the spine, two for the neck, one for the head and then a joint at the top of the head (purely for aesthetic reasons). Now if you need to reposition the joints, you can rotate them and use only the Translate X attribute of each joint (which refers to its length). Once you are happy with the joints, select the root joint of the chain and go to Modify>Freeze Transformations. Make sure all joints have 0 values for their rotations, and then go through and name the joints accordingly. Our joint chain is labelled like so from root to tip: "spine1_jnt", "spine2_jnt", "spine3_jnt", "spine4_jnt", "neck1_jnt", "neck2_jnt", "head_jnt" and "headEnd_jnt".

03 Arm and finger joints In the Front view now, create a six-joint chain. Make the first click at the clavicle, following that the shoulder, then the root of the humerus, the elbow, the wrist, and finally at the end of the palm. Again, use just the rotation attributes and Translate X to get the joints into the correct position. Once in place, remember to go to Modify>Freeze Transformations to zero out the rotation values. For the fingers, create the joint chains from the Top view. For each finger, we created a five-joint chain. The first joint was at the origin of each metacarpal, then we created a joint for each of the phalanges, and finally a joint at the tip of the finger. Once you have all your joints in place, select the root joint for each finger and parent them under the wrist joint. Make sure to then go in and rename the joints. As this is for the left arm, also add a prefix of l_ to all the joints.

04 Create the root and hip joint Let's create the root joint next. This joint will carry the entire rig. For the root joint, jump into the Side view and create a single, isolated joint. Create this away from the rest of the rig so it does not attach itself to the current skeleton. Next, hold down the V key and point snap it to the same position as the root of the spine chain. Rename this joint "root_jnt". Now for the hip: still in the Side view, create a two-joint chain, and then rename the joint chain from root to tip: "hip_jnt" and "hipEnd_jnt". Now select hip_jnt and point snap it to the same position as root_jnt. Next take the root joint of the spine (spine1_jnt) and hip_jnt, and parent them under root_jnt. Then take the root joint for the arm chain (clavicle_jnt) and parent it under the closest spine joint (spine4_jnt).

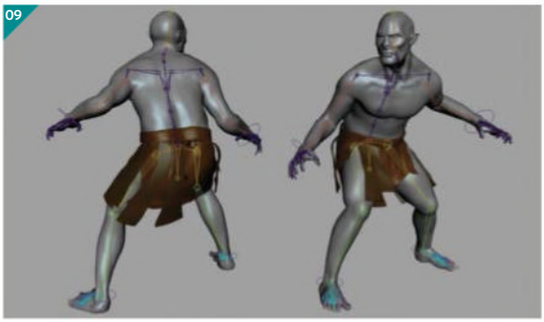
05 Continue to make the leg and toe joints For the leg, create a five-joint chain in the Side view. The first joint was placed just below the root joint, then at the knee, the ankle, the ball of the foot and then the tip of the toes. Again, rename the joints (adding l_), and work in all the views to translate and rotate the joints into place. For the toe joints, create them in a similar way to the finger joints by starting with the metatarsals and then working down the toes. Once the toe joints are created, select the root joint for each toe chain and parent them all under ball_jnt. Remember to freeze the transformations on all the joints so that we have clean rotation values. Now take the root joint for the leg and parent it under hip_jnt. Next, mirror the left joints over to the right-hand side. Start by selecting the leg joint and go to Skeleton>Mirror Joint (Options). When the window pops up, make sure Mirror across is set to YZ, Mirror function is set to Behavior, Search for is l_ and Replace with is r_. Once the leg is mirrored over, repeat the process for the arm. The model is not fully symmetrical so just go in and check that the mirrored joints are correctly in place.

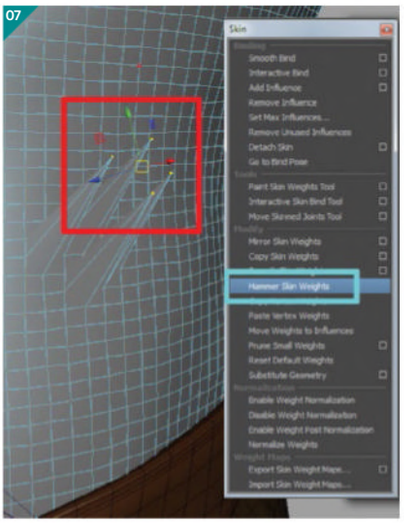
06 Skin the model We now need to attach it to the model, via a process referred to as skinning. In the Outliner, start by selecting all the joints other than the end joints (as we do not want these as part of the bind). Then Shift-select the model in the viewport and go to Skin>Smooth Bind (Options). Use the following settings and hit Bind Skin: Bind to: Selected Joints, Bind Method to Geodesic Voxel (or Classic Distance if you are using Maya 2014, 2013), Normalize weights to Interactive, Max Influences to 8, Falloff to 0.3 and Resolution to 1024. Once the bind has been calculated, you can rotate the joints and the model should deform. Some areas by default will be pretty pants and we'll need to clean up the initial bind. To do this, we primarily use the Paint Skin Weights Tool. With the model selected, activate the tool and in the Settings, you should see a list of joints that are influencing the model. Here you can select a joint and then define how much influence that joint has over the vertices of the mesh. Go through and pose the character by rotating the joints, and use Paint Skin Weights to tidy up the troublesome areas. Only focus on one side as we can always mirror the weighting over. For this tutorial, it took around 45 minutes to one hour to paint the weights.



Export a skeleton to Mudbox

For this tutorial, we exported an OBJ file and added the pose as a sculpt layer in Mudbox. Another technique you may want to try is sending the entire Maya skeleton over to Mudbox. To do this in Maya, group the rig and the geometry all together. Then export the group node out as an FBX file. Now import that file into Mudbox, jump into the Pose tools, and you should see your skeleton in place and ready to pose. Simply subdivide the model, pop the Displacement map on top, and you can handle all your posing in Mudbox using the skeleton created in Maya.





07 Mirror the weights and clean them up With the main forms deforming well on one side, we can mirror the weights over. To do this, select the model and go to Skin>Mirror Skin Weights (Options). In here, set the Mirror across to YZ, enable Direction, and hit Apply. Again, rotate some joints to check if the weights have been mirrored over. At this stage, you'll probably find that you have a few vertices that are not behaving as well as you'd like. For example, on our rig, rotating the arms resulted in some of the vertices of the belly being pulled. To fix this, we could continue to paint the weights, but to fix this quickly we used the Hammer Skin Weights tool. To use this tool, select the troublesome vertices and then go to Skin>Hammer Skin Weights. Hopefully by now, you'll have taken care of the majority of the skinning. Once you've finished using the Hammer Weights tool, pop the rig back to its default pose and mirror the weights over again.

08 Add IK handles and controls At this stage, we could pose the mesh and send it back to Mudbox. What we'd like to do now, however, is quickly add some IK handles and some controls to make the process of posing the character slightly faster. This will also help to keep the legs planted on the ground when posing the hips and the torso. We'll explain the process for the left arm here, but you can repeat the step on the right arm and the legs. Start by going to Skeleton>IK Handles (Options) and then set the Current solver to Rotate-Plane Solver. Next make a first click on l_upperArm_jnt, and then a second click on l_wrist_jnt. You should now be able to select the IK handle that has been created, and using the Move tool, you can pose the entire arm. You can also use the Twist attribute that lives on the IK handle to position the elbow. IK handles are created in world space so you cannot simply zero out the Translate values to get them back to their creation position. To enable us to do this, let's create a control. First, go to Create>NURBS>Circle (make sure Interactive Creation is disabled) and then rename the circle "l_arm_ctrl". Now with l_arm_ctrl selected, hit Cmd/Ctrl+G to group it to itself, and rename this "l_arm_ctrl_offset". We now need to position the control so select l_arm_ctrl_offset, hold down V, and point snap the _offset node to the same position as the IK handle. Now take the IK handle and parent it under l_arm_ctrl. You should now be able to use the control to position the IK handle, and zeroing out the values of the control (not the scale) should result in the IK handle going back to its default pose.

09 Pose the sculpt Using a combination of the FK joints and the IK controls, pose the character as you wish. If you do find that there are artefacts in the pose, do not worry yourself, we will clean these up before we send the model over. With the sculpt in the pose, duplicate the mesh and then hide the original skinned model. Now take the duplicated mesh and go to Edit>Delete By Type>History and then unlock the transform attributes in the Channel Box. Next, select the mesh and hit F3 to switch to the Polygons menu set. Then go to Mesh Tools>Sculpt Geometry Tools (Options). Set the Operation to Relax and then start cleaning up the troublesome areas using the left mouse button. You could handle this in Mudbox but the Relax feature of this tool is a firm favourite of ours.

10 Send the pose to Mudbox Go to File>Export and save the model as an OBJ file. We've named the file "orc_poseA.obj". Next, jump into Mudbox and make sure the sculpt is at its lowest subdivision level. Then go to the Layers tab and make sure you are looking at the Sculpt layers. Click on the Options icon (small, round grey icon), and go to Import Layer. Select orc_poseA.obj and the sculpt should immediately jump to the new pose. You should now be able to jump up the subdivision levels to reapply the hi-res details. If for any reason, you find that the hi-res details have a bubbly effect (this seems to happen in around one in ten sculpts), this is how you can go about cleaning it up. First, pop the sculpt back to its default pose, and then extract a Displacement map (usually a 4k, 32-bit floating point EXR). The mesh we're using has UVs but if it did not, then take advantage of Ptex. Next, delete all the subdivision levels on the sculpt (Mesh>Delete Highest Level). Then resubdivide the mesh back up to its highest level, and apply the Displacement map (UV & Maps>Sculpt Using Map). You should now be able to apply the pose or any other pose and have everything look as it should.

AFTER EFFECTS

Animate a liquid metal logo

After Effects
Chris McFall
www.unitedfilmdom.com

Chris is a filmmaker based in Cardiff, UK. Trading under the moniker of United Filmdom, he is just as happy creating 3D worlds as he is filming our own.

If you grew up in the Eighties and Nineties, you'll remember the mind-blowing liquid-metal effects that made an entire generation believe in Skynet. In my mind, liquid metal has always been the pinnacle of cool in VFX, but it's a tricky thing to tame. Fluid simulations, effectors and containers make for unruly companions, and creating such effects has long been out of reach for most of us – until now.

Using After Effects' built-in tools, we can create a great-looking 2D animation, forming logos or any shape we want from running beads of liquid metal. With the help of Mettle's FreeForm Pro plug-in, we can take our animation to the next level, morphing it into a three-dimensional liquid-metal work of art.

The core technique is to create a flowing series of ellipses using the built-in CC Mr Mercury effect, and combining it with a Write-on technique that builds under a track matte of your logo. Through some clever nesting of compositions and a nifty use of layer styles to create a Displacement map, we can prepare a comp that is perfect for use with FreeFoarm Pro. This will provide a clean 3D displacement and a neat Reflection map to get that realistic appearance we want.

This project can be pretty tricky, and it's much easier to explain by demonstration, so be sure to check out the video that accompanies the tutorial. You don't necessarily need to use this technique to create liquid metals – you could just as easily make ink, blood, or even oil.

> "Through some clever nesting of compositions and a nifty use of layer styles to create a Displacement map, we can prepare a comp that is perfect for use with FreeFoarm Pro"

YOUR FREE DOWNLOADS
from **filesilo.co.uk/bks-798**
- Tutorial screenshots
- Video tutorial

01 Prepare your ideas Start your logo as a single vector shape in Illustrator. In After Effects, create a new comp called 'Logo' and paste your logo from Illustrator onto a Solid in your Logo comp. Next, import some textures. You'll need an HDR map, concrete for the background and a brushed-metal texture, which should be pre-comped and named 'Mercury_writeon'. Nest the Logo comp into a new comp called 'Incoming_Metal', then create a Null and use Motion Sketch to draw the liquid's path inside your logo. Now hide the Logo layer.

02 Create the beads Create a Solid called 'Mercury_incoming', add the Mr Mercury effect and link the Producer position to the Null position. This should mean that we're emitting mercury from the Null. Tweak the settings to create finer beads, ensuring that they don't disappear until off canvas. Add Turbulent Noise above the Mr Mercury effect, and tweak it until it looks like chrome. The next step is to copy the Null layer and paste into a new comp called 'AllMetal'. Go back to the Mercury_incoming comp and time-reverse the Null. Keyframe the mercury's Velocity and Birthrate to 0 around the two-second mark and reach its full value at nine seconds.

03 Set up the comp Drag an instance of Logo into the AllMetal comp and stack it above the Mercury_writeon layer. Set the Mercury_writeon TrkMatte to Alpha. Next, add Mr Mercury again, linking the Producer to the Null position, then tweak the setting to suit your scene. I used a Velocity of -0.2, a Birthrate of 1.5, a Longevity of 20 and Gravity set to 0. Assign Resistance a rate of 0.22, set Animation to Jet, set Influencemap to Constant Blobs, Birthsize to 0.09 and Deathsize set to 1.98. Again, add a Turbulent Noise effect before the mercury effect, set as before. Add and keyframe a Vector Blur to the Logo layer so that it goes from soft to defined at around the five-second mark. Drag an instance of the Incoming_metal comp under these layers, and time-reverse it. Now the beads of metal are being sucked into the Null position and the logo is being built behind it.

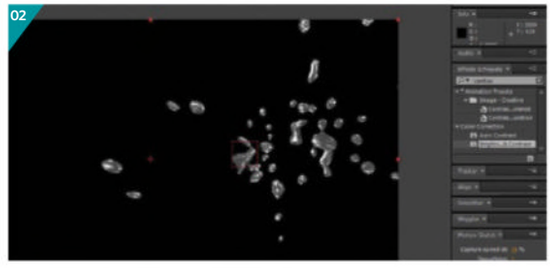
04 Make it stand out Drag AllMetal and your background texture into a new comp named 3D_comp, then into the AllMetal layer. Add Bevel and Emboss set to Soft Chisel and scale the effect to get a fine ridge. Next, add a slight Drop Shadow. Drag 3D_comp and your HDR_Map into a new comp named Main_comp, then add a camera and lights. Drag another instance of AllMetal into another comp called Displacement_map. Create a black Solid under the AllMetal layer, setting Trkmatte to Alpha. Set the Layer Style to Inner Glow, the Technique to Precise and adjust the Size and Range to suit your animation. Add an adjustment layer with Fastblur set to 3 and Invert.

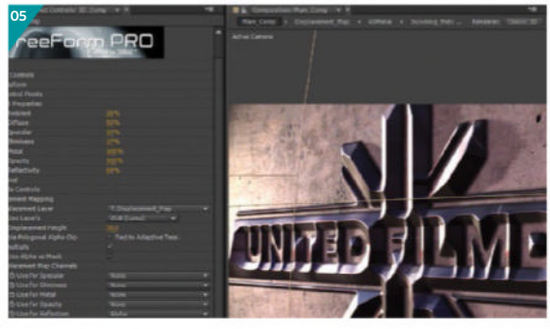
05 Adjust the lights In the Main_ comp, keyframe your camera move, then add the third-party FreeForm Pro effect to the 3D_Comp layer. Set FreeForm's Displacement Mapping layer to Displacement_map. We will use RGB (luma) Layers set to a Height of 30. Change Tessellation To Subdiv to 1,000, then set the Displacement Map Channels>Use For Reflection to Alpha. Set the Reflection Mapping layer to our HDR map, then adjust this mapping according to the style of map that you're using. Next, set Blending to Hard Light. Feel free to move the lights to cast long shadows from the logo, tweaking the Freeform's Material Properties.

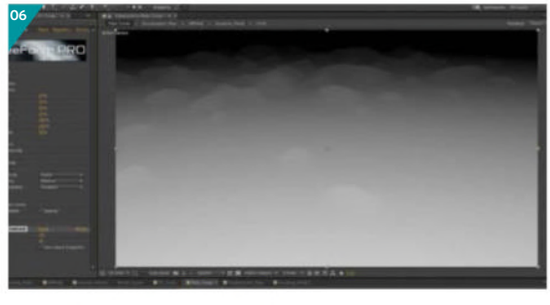
06 Finish up For a realistic depth of field, go to FreeForm's Render mode and set to Depth, adding the Brightness/Contrast effect. Avoid keyframing the tweak to ensure that the depth is visible throughout. Render and import this pass, then add it to the Main_comp and make it invisible. Change the Render mode back to Full and add a Camera Lens Blur effect, changing the layer to Depth_map and the Blur_radius to 86. It's best to keyframe the Blur Focal Distance throughout. To garnish, add adjustment layers with blur, flare and colour grading.

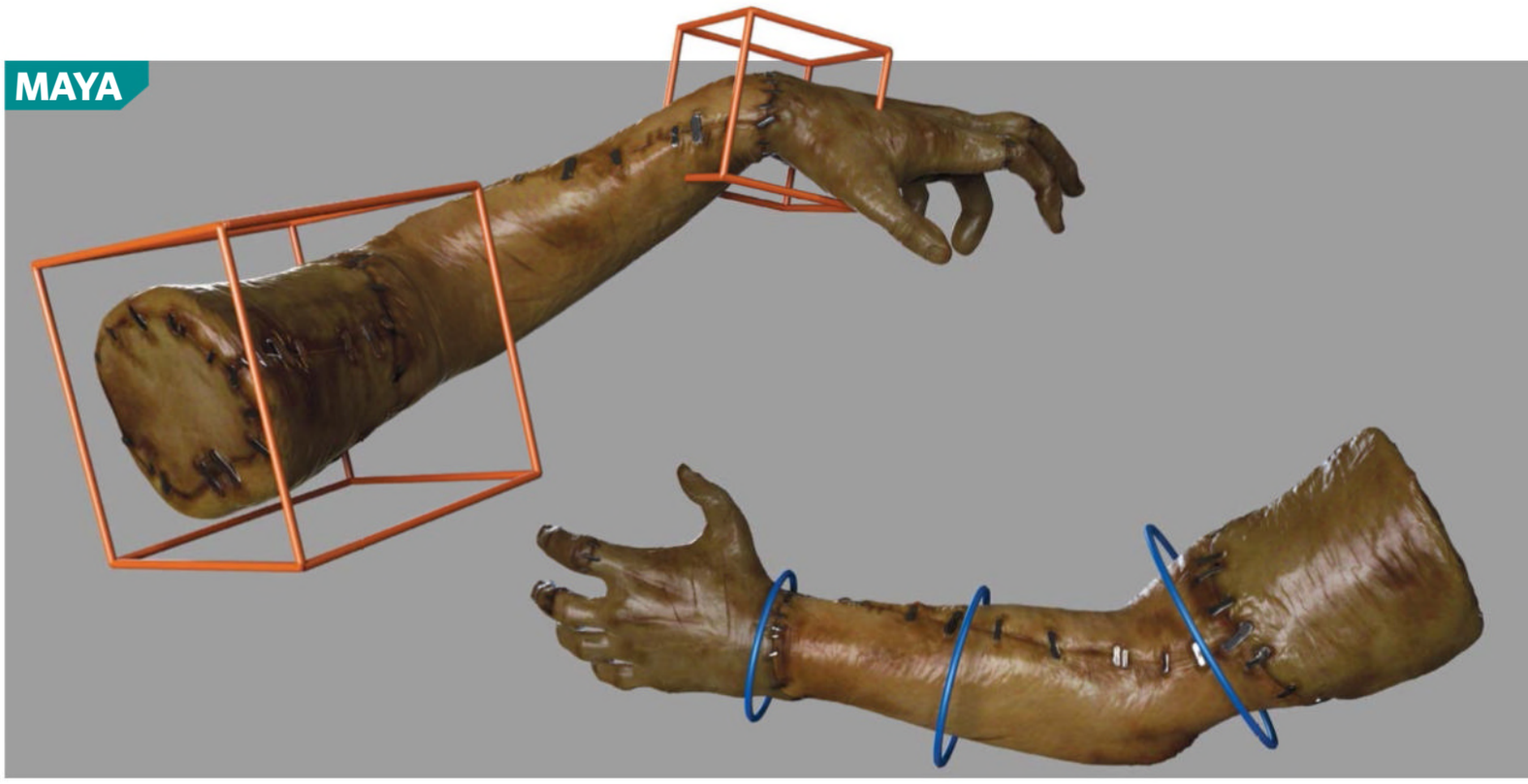
Double up

Do you need multiple segments of your logo to be built simultaneously? Would you like the metal to flow down two paths simultaneously? In the Project window, select the Incoming_metal comp and press Cmd/Ctrl+D to duplicate. In this duplicate, adjust the Motion Sketch path. Add this to the AllMetal comp and copy and paste the Null with the new motion path with it (but make sure this copy isn't time-reversed). Duplicate the Logo and Mercury_writeon layers in the comp. Connect the Mercury_writeon layers with the Mr Mercury producer to this new Null's position. Repeat this as many times as necessary.









Rigging with FK and IK

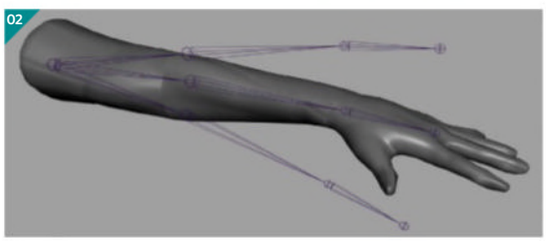
Maya
Jahirul Amin
jahirulamin.com

Jahirul Amin is a 3D trainer at Double Negative and has a passion for rigging and animation

Here we will look at an efficient technique of rigging an arm that enables both FK and IK movement. A common way of handling such a task would be to create three joint chains: an FK, an IK and a bind chain (this is the chain that is used to deform the model), and then constrain the bind chain to both FK and the IK chain. From here you would then use Set Driven Keys to control the weighting attribute of the constraints, enabling you to control which chain (the FK or the IK) should drive the bind chain. Now there is nothing really wrong with this method, but in this tutorial we'll do something slightly different in that rather than using constraints, we'll use blendColor nodes instead. The advantages of this method is that it can reduce set-up times, and in more complex setups it can also provide a speedier rig.

For this tutorial, we've included a model of an arm, so feel free to use that or work on your own model. Before you start rigging, however, do examine the model. Make sure the Translate and Rotate values are at 0, and the Scale values are at 1. You should also delete any history off the model by going to Edit> Delete By Type>History. After that, you should be good to go. We're using Maya 2016 here, so if you are using Maya 2015 or 2014, you'll find that some of the buttons may be in a slightly different place. The concepts should still work though, so no quarrels there. Next issue, we'll look at how to use Set Driven Key to rig the fingers.

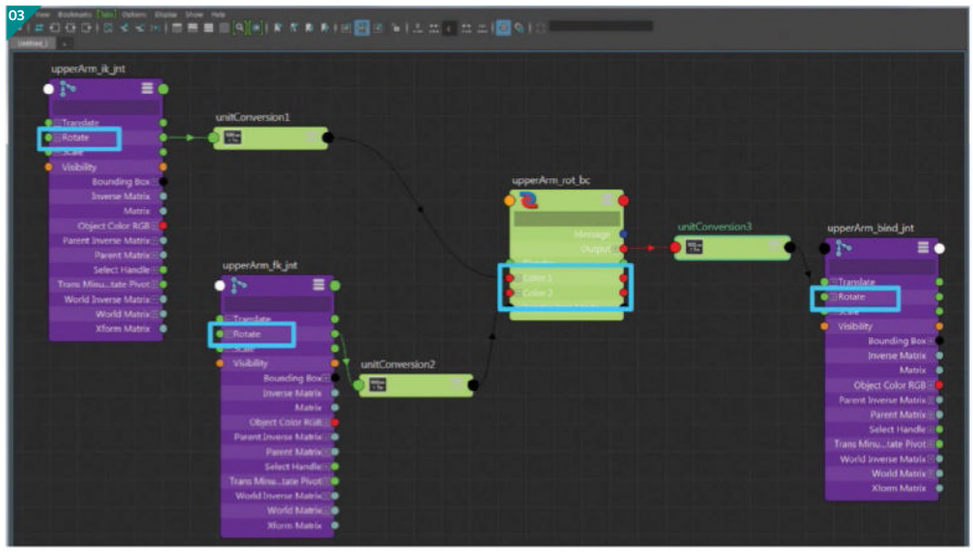
01 Create the arm joints Open up scene file 00_start. First create the arm joint chain. In the top view, go to Skeleton>Joint Tool and draw a four-joint chain by clicking first at the root of the arm, then at the elbow, next at the wrist and finally at the end of the palm. Add a slight bend to the arm as this makes it easier for the IK solver to figure out how the joint chain should articulate. Now from root to tip, rename the joints like so: 'upperArm_jnt', 'lowerArm_jnt', 'wrist_jnt' and 'wristEnd_jnt'. Check the joints from all angles and examine their placement. The Perspective view will be ideal for this. We ended up translating upperArm_jnt so that the arm chain sat in the middle and rotated the joints and used the Translate X attribute on all the children joints to get them into place. Once you are happy with the joint placement, select upperArm_jnt and go to Modify>Freeze Transformation. This will clean out all the values in the Rotate channels for all the joints. Next, we want to have the flexion of the arm driven by positive Z rotation so select upperArm_jnt and go to Skeleton>Orient Joint. When the dialog box pops up, set the Primary Axis to X, the Secondary Axis to Y and the Secondary Axis World Orientation to Z (+). Check the rotations of the joints to see if you are happy with the results.

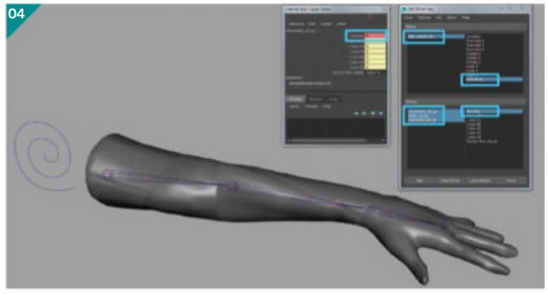


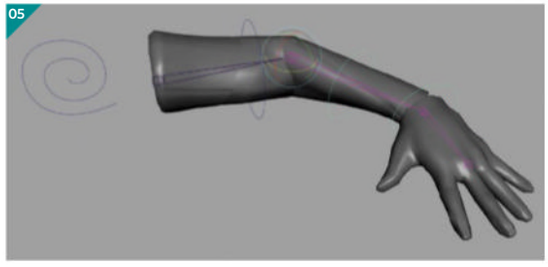
YOUR FREE DOWNLOADS
from **filesilo.co.uk/bks-798**
- Tutorial screenshots
- Video tutorial

02 The FK and IK joint chains We'll now create the FK and the IK joint chains. Select upperArm_jnt and hit Cmd/Ctrl+D to duplicate it. Then with the upperArm_jnt1 (the duplicate chain) selected, go to Modify>Search and Replace Names. In the Search for field, type _jnt and then in the Replace with field, type _fk_jnt. Then delete the one that is on the end of upperArm_fk_jnt. Next, duplicate upperArm_jnt again, only this time, rename the joint chain to end with '_ik_jnt'. Lastly, select upperArm_jnt and use the Search and Replace tool to change the name of the joint chain from _jnt to _bind_jnt. By updating the name of this joint chain, we can easily distinguish the joints that will be used for skinning from those that will not. The additional joints chains we have now (the FK and the IK set) should be sitting in exactly the same position as the original joint chain. The next thing we did was to create some low-res geometry and parent it to the arm. This is so that we could test out how the arm articulates without having to work on skinning the model. You can do this by duplicating the existing arm, reducing the poly count, chopping up the reduced mesh using the Extract tool and then parenting this geometry to the relevant bind joint. Rotating the bind joints should now drive the low-res arm.

03 Drive the bind joint chain We will now drive the bind joint chain using blendColor nodes. In the Outliner, start by selecting upperArm_bind_jnt, upperArm_fk_jnt and upperArm_ik_jnt, and go to Windows>Node Editor. Next, and still in the Node Editor, hit the Tab key on the keyboard and type 'blendcolors', and then hit Enter to create the node. Double-click the blendColors node now to open it in the Attribute Editor then rename it to 'upperArm_rot_bc'. The 'rot' is for rotation and the 'bc' for blendColors. In the Attribute Editor for the node, you will also notice a Blender attribute. This is what we will use later on to drive the switch between FK and IK. Now, in the Node Editor, click on the top-right corner icon for all the joints and upperArm_rot_bc to reveal all the attributes. Next, drag the Rotate output from upperArm_ik_jnt and drop it into Color1 on the upperArm_rot_bc node. Then grab the Rotate output of upperArm_fk_jnt and drop that into the Color2 slot on the upperArm__rot_bc node. Lastly, take the Output of upperArm_rot_bc and drop it into the Rotate input for upperArm_bid_jnt. As we connect the rotate attributes into and out of the blendColors node, you'll notice unitConversion nodes are created. This is all fine so leave them as they are. With all that done. Repeat the process for the lowerArm joint and the wrist joint. For each joint, you will need to create its own blendColor node.

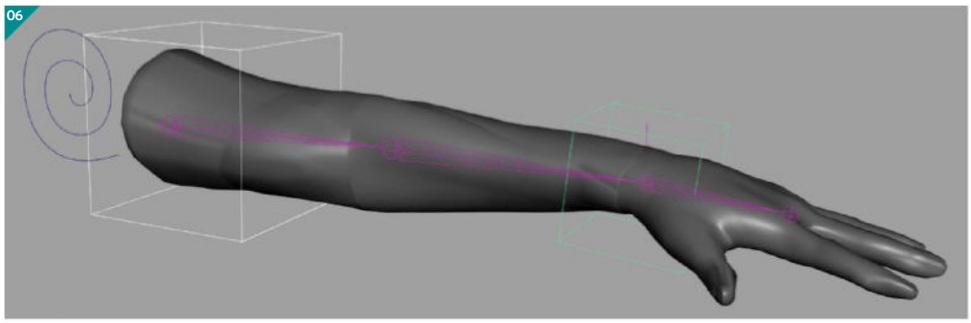


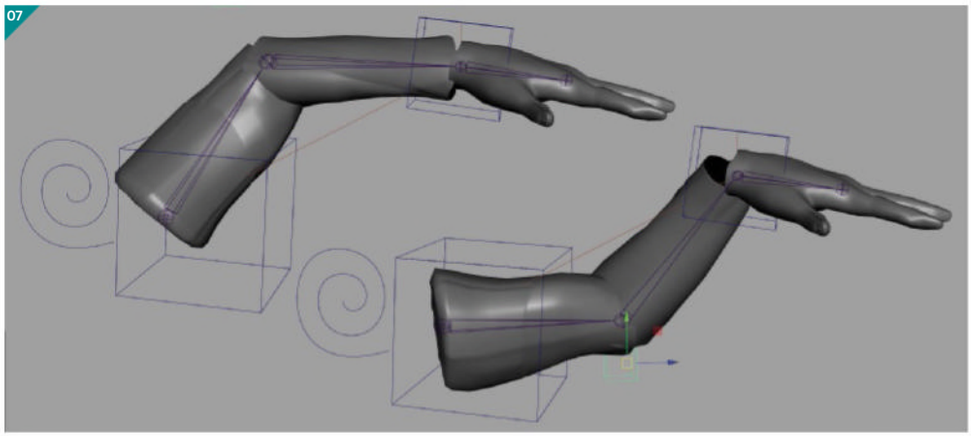


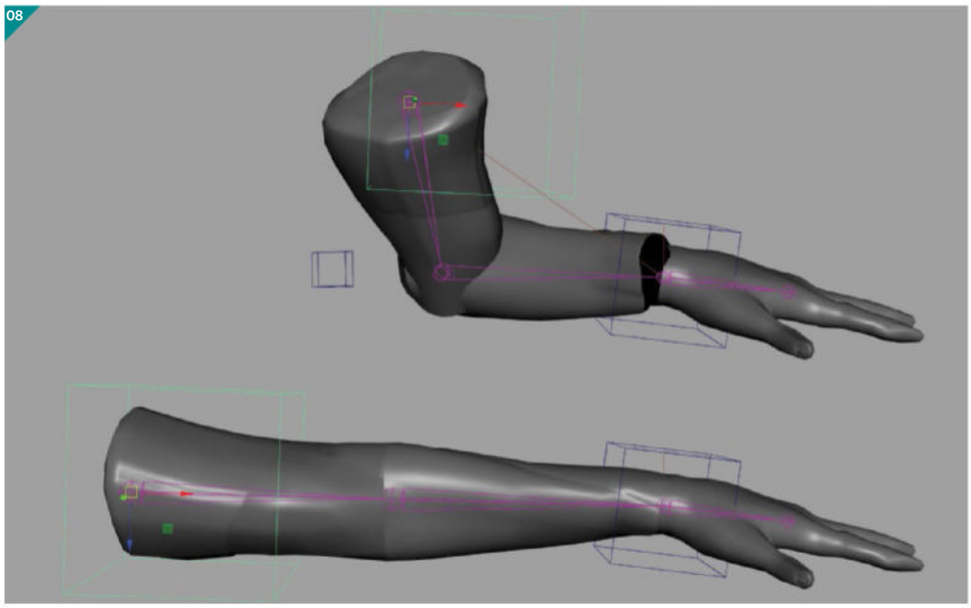


04 Blend between FK and IK Now that we are at this stage, we'll need a control with an attribute that will let us switch between the FK and IK. For this control, simply use the CV Curve Tool to create a swirl shape. Then position the control close to the upper arm and go to Modify>Freeze Transformations and Modify>Center Pivot. Next, rename the new curve to fkIk_switch_ctrl. Now, with the control selected go to Modify>Add Attribute. When the window pops up, give it the following name: 'arm_fkIk'. Set the Data Type to Float. Set the Minimum to 0 (this will be FK mode), the Maximum to 1 (this will be IK mode) and the Default at 0. Hit OK when you have filled in all the fields and you should see this new attribute in the Channel Box. Now we'll use Set Driven Keys (SDK) to drive the blend between FK and IK. First, go to Windows>Rendering Editors>Hypershade and jump to the Utilities tab. From here, select the three blendColor nodes that you should have created and then with the Animation menu active, go to Keys>Set Driven Key>Key. This should load the three blendColor nodes and drop them into the Driven section of the SDK window. Now select fkIk_swtich_ctrl and hit the Load Driver button. Next, set the arm_fkIk attribute to 0 and then set the Blender attributes on all three blendColor nodes to 0. Back in the SDK window, hit the Key button. Following that, set the arm_fkIk attribute to 1, the Blender attributes to 1 and hit Key once more on the SDK window. If you rotate the FK and IK arm joints around now, you should see that they drive the bind chain and you can switch between the two modes.

05 Create the FK controls Let's create the controls for the FK joint chain. Start with a NURBS circle in Create>NURBS Primitive and rename it 'upperArm_fk_ctrl'. With this still selected, hit Cmd/Ctrl+G to create two group nodes that will live above it in its hierarchy. Rename the group nodes from top-down 'upperArm_fk_ctrl_offset' and 'upperArm_fk_ctrl_sdk'. The _offset group will be used to place the control correctly and the _sdk group can be used for more control. Next select upperArm_fk_ctrl_offset and parent it under upperArm_fk_jnt. You will see some values pop into the Channel Box for the _offset group. From here, zero out all the values for the Translate and the Rotate channels (not Scale). The control should now pop to the correct place. Unparent the _offset group to bring it out of the joint chain hierarchy. To edit the shape of the control, jump into component mode and use the CVs. Scale the CVs up and move them down the arm slightly. Now, select in this order: upperArm_fk_ctrl, upperArm_fk_jnt and go to Constrain>Orient. Repeat the process for lowerArm_fk_jnt and wrist_fk_jnt. You should now have three FK controls. Next, parent one control under the other so that we can create a forward kinematics behaviour. Do this by parenting wrist_fk_ctrl_offset under lowerArm_fk_ctrl, and then parent lowerArm_fk_ctrl_offset under upperArm_fk_ctrl. The FK controls should now be working and driving the FK joint chain. Rotate the controls to test them out. Once you are satisfied, zero out the rotations to return the arm to its default position.







06 Create the IK controls Before we move on to the IK controls, add an IK handle. To do so, switch to the Rigging menu (Animation menu if you are in Maya 2015/14) and go to Skeleton>IK Handle Tool (Options) and set the Current solver to Rotate-Plane Solver. With the tool active now, click on upperArm_ik_jnt and then on wrist_ik_jnt. Now select the newly created IK handle and rename it arm_ik. Also, look for the effector node (living underlowerArm_ik_jnt) and rename it 'arm_ik_effector'. Next, we need to create a control for the IK handle. You can use any NURBS curve you want. Take the new curve shape, rename it 'arm_ik_ctrl', and then with the control selected, hit Cmd/Ctrl+G twice. Rename the top group in the hierarchy 'arm_ik_ctrl_offset' and the group below 'arm_ik_ctrl_sdk'. To position the control (as our IK handle is just translated in World mode), select arm_ik_ctrl_offset, hold down the V key and then snap the position of our _offset group to the IK handle. We now want the orientation of the control to match the orientation of the wrist joint. To achieve this, select wrist_ik_jnt, Shift+select arm_ik_ctrl_offset and go to Constrain>Orient (in the Options make sure Maintain Position is disabled). Now search for the orient node in the Outliner (under arm_ik_ctrl_offset) and delete it. Next duplicate arm_ik_ctrl_offset, and rename the two groups and the ctrl 'arm_root_ik_ctrl'. Select the _offset node for this new control, parent it under upperArm_ik_jnt, zero out the Translate and Rotate values and then unparent it. You should now have two controls in place for the IK arm: one at the root and one at the wrist. If you need to reshape the controls, make sure you do so in component mode.

07 Create the IK elbow control Set the arm to be driven by the IK joints (using our custom attribute) so we can see how it behaves. Then select in this order: arm_ik_ctrl, arm_ik, and then go to Constrain>Point. Then select in this order: arm_ik_ctrl, wrist_ik_jnt, and go to Constrain>Orient. Lastly, select in this order: arm_root_ik_ctrl, upperArm_ik_jnt and go to Constrain>Point. Our controls should now drive the arm. You'll notice that the root control does not carry the bind joints – we'll fix this later. Now for the IK elbow control select arm_root_ik_ctrl_offset and hit Cmd/Ctrl+D to duplicate it. Then use the Search and Replace tool to rename it from '_arm_root_ik' to '_elbow_ik', and delete the one that will be hanging on the end of the offset group. To position the new control, select in this order: upperArm_ik_jnt, lowerArm_ik_jnt, wrist_ik_jnt, elbow_ik_ctrl_offset and go to Constrain>Point. Next, select in this order: lowerArm_ik_jnt, elbow_ik_ctrl_offset and go to Constrain>Aim. Look under elbow_ik_ctrl_offset now and delete the two constraint nodes we just created. Now select elbow_ik_ctrl_offset and translate the control slightly away from the arm using the x axis (make sure to do this in Local mode). Then select in this order: elbow_ik_ctrl, arm_ik and go Constrain>Pole Vector. The new control now orients the elbow. You can also add a custom attribute on arm_ik_ctrl to drive the Twist value on the arm_ik. for a second method of twisting the arm.

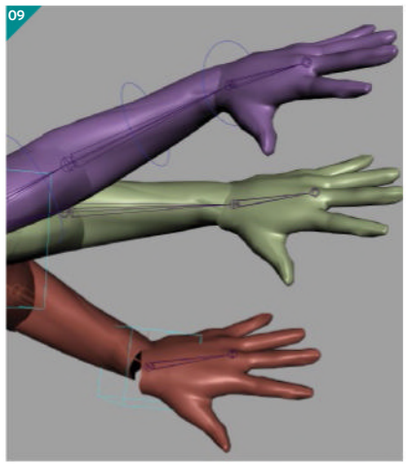
08 Translate the root We now want to drive the position of upperArm_bind_jnt with its FK and IK counterparts. Bring upperArm_bind_jnt, upperArm_fk_jnt and upperArm_ik_jnt into the Node Editor. Create a new blendColors node and rename it 'upperArm_trans_bc'. Plug the Translate output from upperArm_ik_jnt into the Color1 input on upperArm_trans_bc and then the Translate output of upperArm_fk_jnt into the Color2 input. Feed the output of upperArm_trans_bc into the Translate of upperArm_bind_jnt. Use SDKs to drive the blender attribute on upperArm_trans_bc to enable the switch between FK and IK modes. The driving attribute should still be arm_fkIk. The bind arm should follow the IK arm. To have it follow FK, select in this order: upperArm_fk_ctrl, upperArm_fk_jnt and go to Constrain>Point.

09 Control colour and visibility

Time for some rigging housework. First we'll start with visibility so hide the FK controls when we are in IK mode and hide the IK controls when we are in FK mode. Start by selecting all three IK controls and upperArm_fk_ctrl. Then go to Key>Set Driven Key>Set to bring them as driven objects into the SDK window. Select fkIk_switch_ctrl now and hit the Load Driver button. Also set the arm_fkIk attribute to 0. Select all three IK controls now and set the Visibility attribute to 0. Back in the SDK window, highlight arm_fkIk in the top-right box, all four objects in the bottom-left box, the Visibility attribute in the bottom-right box and hit Key on the SDK window. Then set the arm_fkIk attribute to 1, enable the Visibility to all the IK controls, disable the Visibility on upperArm_fk_ctrl and hit Key once more. The next step is to colour the controls. To do this, select them one at a time and in the Attribute Editor, scroll down to the Display>Drawing Overrides section. From here, check the Enable Overrides option and then pick a colour using the Slider below or set the mode to RGB (2016 only) to select any colour of your choice. The next thing you could do is create some geometry for the IK and the FK arm chains (just duplicate the original lowRes geo) and then parent it under the relevant joint chain. You can also add shaders to each set of geometry (we just added Blinn shaders) so you can quickly distinguish between the two modes. Another thing we did was to add two custom attributes to the fkIk_switch_ctrl that would let us show and hide the visibility for the FK and IK arm geometry.

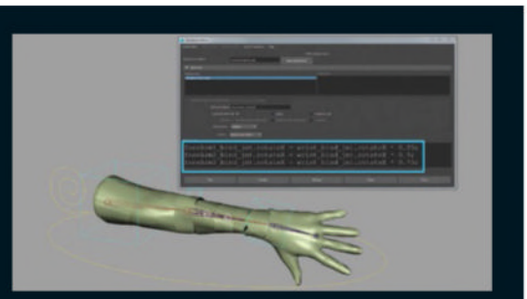
10 Global SRT and cleaning up

Let's create a control that will carry the entire rig. Start with a NURBS circle and rename it 'arm_global_SRT'. With the control selected, hit Cmd/Ctrl+G twice and rename the group 'nodes _offset and _sdk'. Next, select the CVs of the curve and scale them up so the control surrounds the entire arm. It's time for some cleaning up. Take arm_root_ik_ctrl_offset, arm_ik_ctrl_offset, elbow_ik_ctrl_offset and hit Cmd/Ctrl+G. Rename this group 'arm_ik_ctrl_grp'. Next take arm_ik and upperArm_ik_jnt and group them together. Rename this group 'to_ik_rig_grp'. Then take upperArm_fk_jnt and hit Cmd/Ctrl+G. Rename this group: 'arm_fk_jnt_grp'. Next take upperArm_fk_ctrl_offset, hit Cmd/Ctrl+G and call this group 'arm_fk_ctrl_grp'. Then, take upperArm_bind_jnt and hit Cmd/Ctrl+G. Rename this group 'arm_bind_jnt_grp'. You should now have five new groups. Select all of them, along with fkIk_swith_ctrl and parent them under arm_global_SRT. Select arm_geo_grp and arm_global_SRT_offset, hit Cmd/Ctrl+G to group them and then rename that group 'char_arm'.



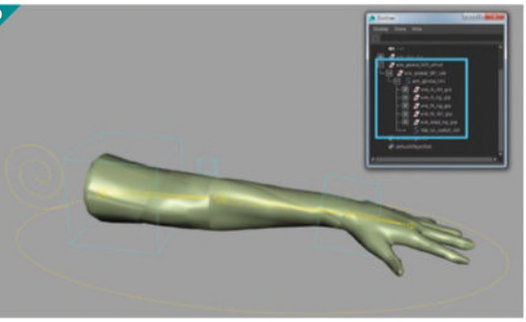
11 Finalise the arm rig

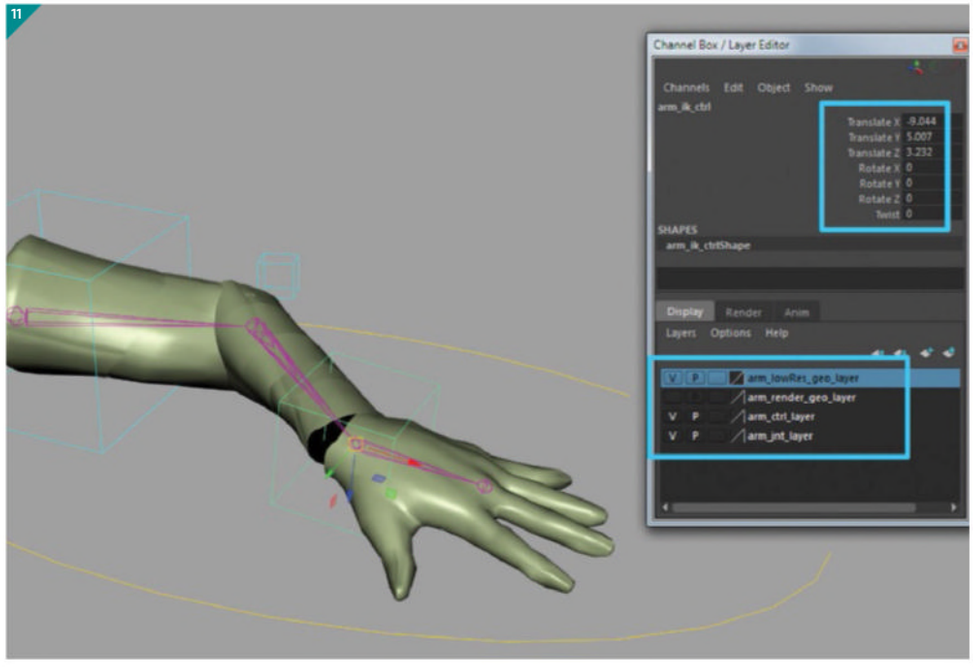
We now want to lock and hide certain attributes on the controls that we do not want the animators to key. Select fkIk_switch_ctrl and in the Channel Box highlight all the Translate, Rotate, Scale and the Visibility attributes, and then click the RMB and go to Lock and Hide Selected. For the IK elbow and root control, lock and hide the Rotate, Scale and Visibility attributes. For the IK wrist control, lock and hide the Scale and the Visibility attributes. For upperArm_fk_ctrl, lock the Scale and the Visibility attributes. And for the remaining FK controls, lock and hide the Translate, the Scale and the Visibility attributes. Then go in and find arm_ik, hit Cmd/Ctrl+H to hide it and then lock its visibility. Next, create some Display Layers to let you quickly show and hide the joints, the controls and the geometry. At this stage, you should now have an arm rig that lets you animate in either FK or IK mode. Next time, we'll work on rigging the fingers. Happy rigging.



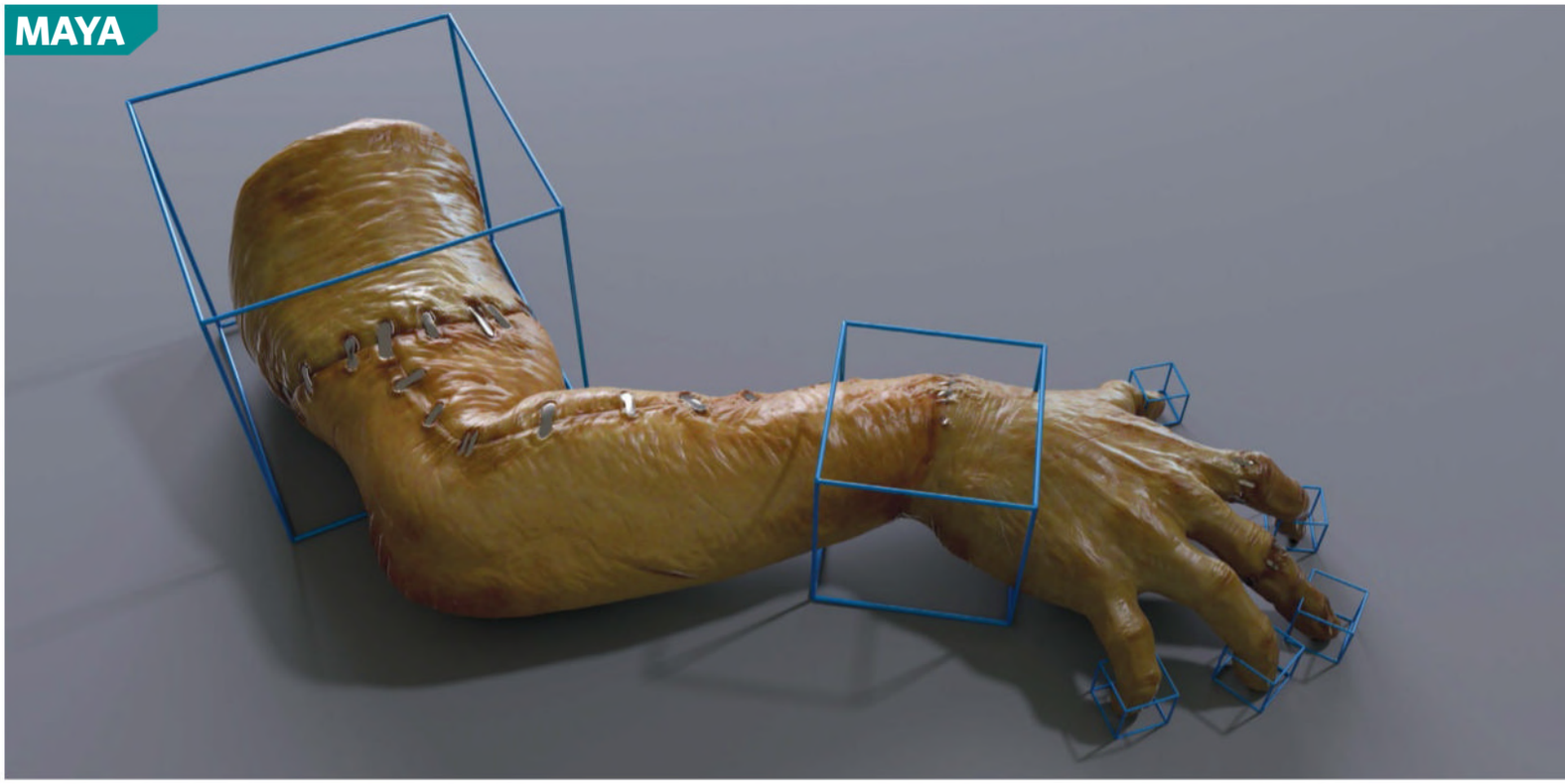
Adding forearm twisting

With the arm ready to go, why not add a few extra joints to the forearm, which would provide better deformation when twisting. Here is a quick overview of how we approached it for this tutorial; the accompanying video on FileSilo goes into more detail so you can re-create it yourself. First, create some additional joints and evenly space them between the elbow and the wrist joints. Then you will need to parent all the additional joints under lowerArm_bind_jnt. After that, use an expression or some nodes to drive the Rotate X of the additional joints using the Rotate X value of wrist_bind_jnt. Make sure that the joints closer to the wrist have more influence from wrist_bind_jnt, as that will help create a more natural falloff.





MAYA



Maya 2016

Jahirul Amin
jahirulamin.com

Jahirul Amin is a 3D trainer at Double Negative and has a passion for rigging and animation

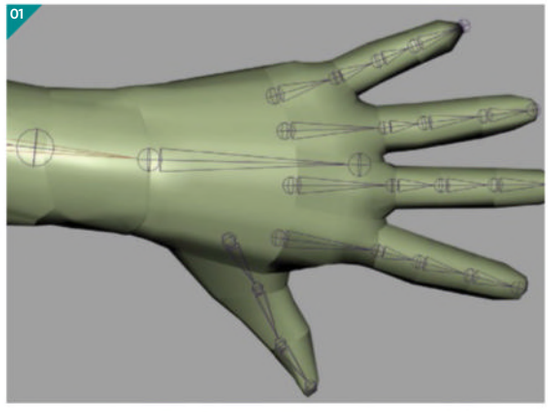
Use Set Driven Keys to rig a hand

Over the next few steps, we will continue working on the arm rig from where we left off last time in Part 1. This time, we will focus on the fingers. We normally use an FK-only based setup and use Set Driven Keys to hit a range of poses quickly, with additional controls to pose each digit of the fingers. On this occasion, however, we will use the fingers to interact with a ground plane (imagine the Thing from the Addams Family). We are going to go for an IK-based setup that then drives an FK rig. This will let us keep the fingertips locked to a certain position but still offset each digit should we wish to do so. We'll also add the functionality that the IK rig can either follow or not follow the main arm rig. That way, if you wish to animate the fingers in FK mode only, the IK won't stand in your way.

We've prepared some scene files for you so that we can kick things off straight away. You'll find the start file on FileSilo in: scenes/02_rigging/part2/00_start.ma. This is pretty much where we left off last time; the only difference is we've gone in and chopped up the low-resolution geometry for the fingers.

01 Create the finger joints Let's get things going by creating the finger joints. Jump into the Front view and create a five-joint chain for the index finger. First click at where you feel the metacarpal joint should go, then at the first knuckle, then at the second knuckle, at the third knuckle, and lastly, create a joint at the tip of the finger. As you create the joints that will sit in the finger, add a slight arc to the joints (like a rainbow) so that the IK handle will be able to pick up the bend point. Next, select the root joint of the new chain and go to Skeleton>Orient Joint (Options). When the dialog box pops up, set the Primary Axis to X, the Secondary Axis to Z and the Secondary Axis World Orientation to Z (-). Positive Z rotation should now be flexing the fingers. Now rename the joint chain from root to tip like so: 'indexA_bind_jnt', 'indexB_bind_jnt', 'indexC_bind_jnt', 'indexD_bind_jnt' and 'index_end_bind_jnt'. Jump into Top view, select indexA_bind_jnt and hit Cmd/Ctrl+D to duplicate the joint chain. Select the root joint of the duplicated chain and translate it for the middle finger. Repeat the process for the ring finger, pinky and thumb. For the thumb, delete the end joint as it has one too many and rename the joints.

01



YOUR FREE DOWNLOADS
from **filesilo.co.uk/bks-798**
- Tutorial screenshots
- Video tutorial

02 Position and rotate all the finger joints

Now that we have all the main joints, start positioning and orientating them into place. Remember that you can translate and rotate the root joint in any axis but for all children joints, you should only use Translate X. You can, however, rotate all the joints on any axis. Once you have positioned all the joints, go to Modify>Freeze Transformations to remove all the rotational values. With the joints in place, select the <finger>B_bind_jnt of each chain and hit Cmd/Ctrl+D to duplicate it. Select all the duplicated joints now and go to Modify>Search and Replace. In here, set the 'Search for' to _bind_jnt and the 'Replace with' to _ik_jnt. You'll also need to go in and remove the one hanging on the root joint for all the IK chains now. With all our joints in place now, take the low-res geometry for the fingers and then parent the relevant piece under the relevant _bind joint, and we will do this so that we can check the behaviour of the rig as we go.

03 Create IK handles and controls

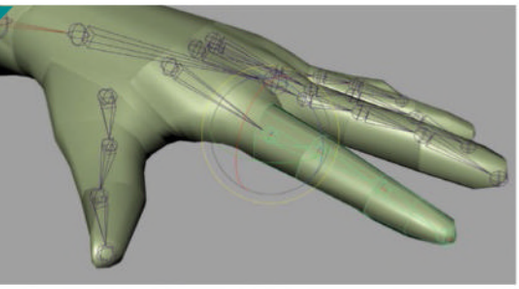
Select all the _bind_jnt chains and then hit Cmd/Ctrl+H to hide them for now so we can focus on the IK chains. For each IK chain, create an IK handle (Rotate-Plane Solver) starting from <finger>B_ik_jnt and ending at <finger>_end_ik_jnt. You should now have five IK handles. Make sure to go in and rename them like so: 'index_ik', 'middle_ik', 'ring_ik', 'pinky_ik' and 'thumb_ik'. Now select them one at a time and test how they affect the joint chain. If you are not happy with them, repeat Step 2 until you are. Next, create a control for each IK handle using the same process that we did in the previous tutorial, or like we have done here, simply duplicate the wrist_ik_ctrl and then use that shape. Here is a quick recap of getting the control to drive the IK handles: create a NURBS shape and rename it appropriately, for example name it like so: 'index_ik_ctrl'. Next, group the curve to itself twice. Then go ahead and add the suffix '_offset' to the top-most group and then '_sdk' to the group above the control curve. Next, move the _offset node to the same positon of the IK handle (in this example this will be: index_ik) it will control, and lastly, point constrain the IK handle to the control curve (index_ik_ctrl).

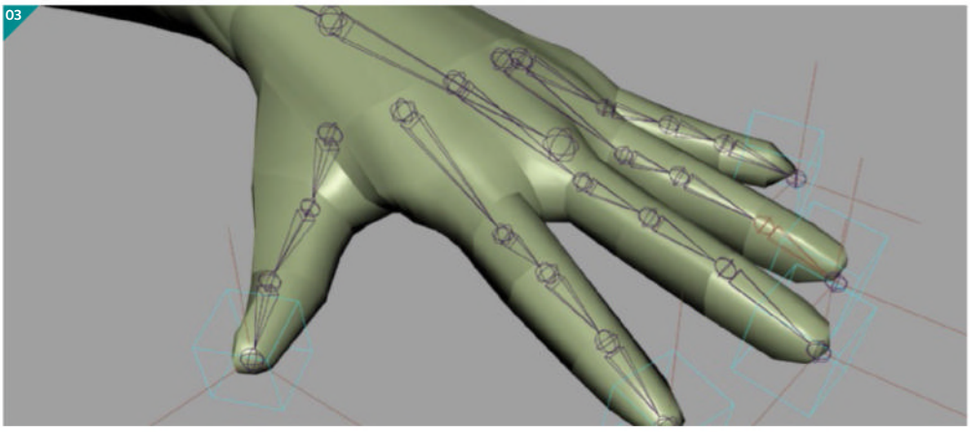
04 Create IK finger twist controls

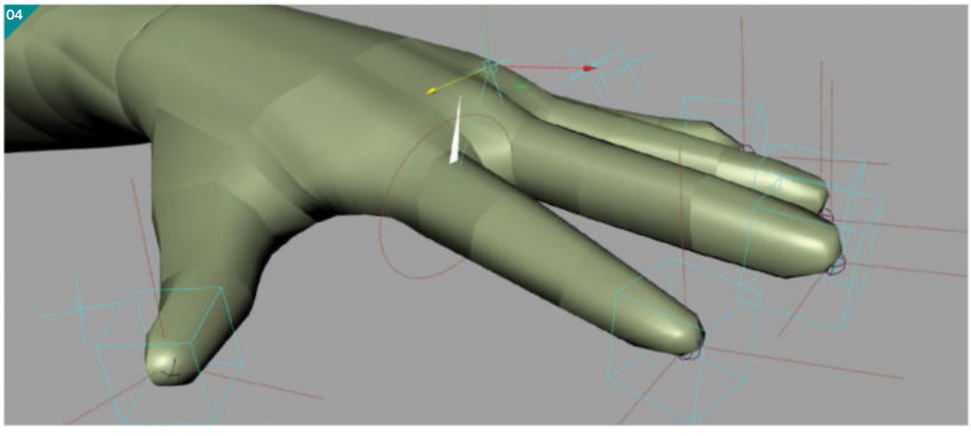
Let's look at adding something similar to enable us to twist the fingers. Start by creating a NURBS curve (of any shape of your choice) and name it 'index_twist_ik_ctrl'. Then create the control hierarchy using group nodes as we have done in the past. Next, we will need to position the control. To do this, all you have to do is just select the following objects in this order: indexB_ik_jnt, indexC_ik_jnt, indexD_ik_jnt, index_end_ik_jnt, index_twist_ik_ctrl_offset, and go to Constrain>Point (make sure Maintain Offset is disabled). Now select in this order: indexC_ik_jnt, indexD_ik_jnt, index_twist_ik_ctrl_offset, and go to Constrain>Aim. With our control correctly aligned, go into the Outliner, select the two constraint nodes under index_twist_ik_ctrl_offset and then delete them. You can now select index_twist_ik_ctrl_offset and in Local translate mode, drag it out slightly so that it can be more easily selected than before. Next, select index_twist_ik_ctrl, then Shift+select index_ik and go to Constrain>Pole Vector. That's one down. Now just repeat this step for the remaining fingers. To add an extra method of twisting the fingers, add the attribute 'twist' to all the IK finger controls (leave the Min and Max blank). If you duplicated the IK wrist control as we have done, then this attribute will already be present on your IK finger controls. Now go to Windows>General Editors>Connection Editor, and add index_IK_ctrl to the left column and index_IK to the right column. Now just highlight 'twist' in both the left and right columns and we should be good to go. Again, you can repeat this step for all the other fingers.

Skinning the arm

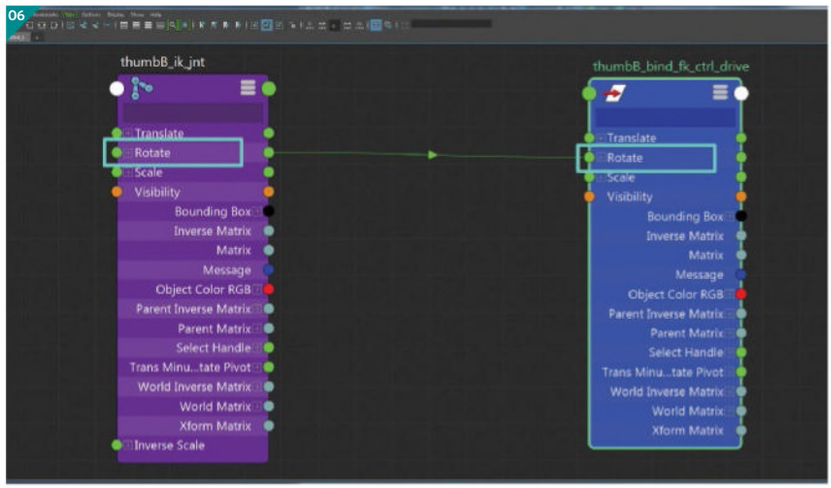
Once you are happy with the rig, you'll want to use it to drive the geometry. You'll want to start by selecting all the _bind joints and dropping them into a Quick Select Set. With the joints still selected, add the arm that we want to bind to the selection and go to Skin>Bind Skin (Options) and use the following settings: Bind to: Selected Joints, Bind method: Geodesic Voxel, Falloff: 0.3 and Resolution: 1,024. All the other settings can stay as they are. Now test out deformations using the rig in both FK and IK modes.

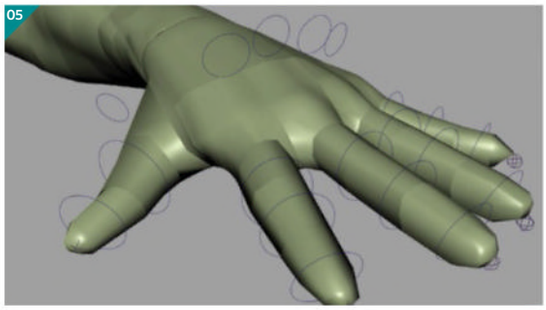


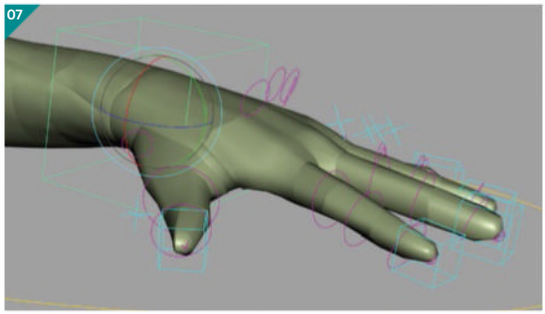


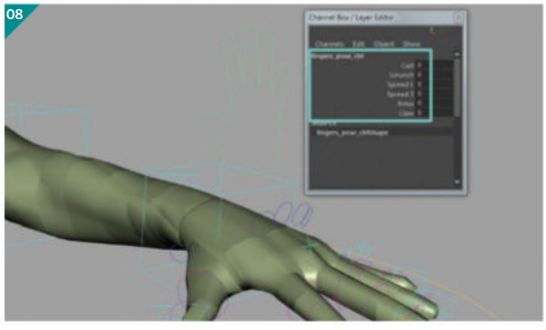


05 FK finger controls Now we'll turn to the FK controls. We've supplied a very simple script that will go in and create the controls for any selected object that has a name ending with _jnt. If you've used a different naming convention throughout your rig, just modify the script (it's pretty self-explanatory where) to make it work. Now, open up the Script Editor and in a new Python tab, click Load Script. Navigate to the scripts directory of the supplied Maya project and load the create_finger_ctrls.py script. By using this script, we will create an additional group within each of the controls' hierarchy structure (_drive). This _drive will be used to drive the FK controls using the IK joints. By doing this on a level above the _sdk group, we will still be able to create a range of poses with the fingers that the animator can quickly hit. To run the script, select all the _bind_jnt (excluding _end_jnt), hit Cmd/Ctrl+A in the bottom half of the Script Editor and then hit Enter to Execute it. For each finger joint now, you should see a control. If they are hard to select, use Component mode to play with their shapes. One thing the script did not do, however, is create the parent-child relationship between each control to mimic the FK hierarchy of our joint chain. We'll have to do that manually. For the index finger (you can repeat this for all the other fingers) select indexD_fk_ctrl_offset and parent it under indexC_ctrl. Then take indexC_fk_offset and parent it under indexB_fk_ctrl. And lastly, take indexB_fk_offset and parent it under indexA_fk_ctrl. The FK controls will now drive the rotations of the joints, they will not, however, translate the root joint of each finger. For this to happen, select in this order: index_A_fk_ctrl, indexA_bind_jnt and go to Constrain>Point. Now all you have to do is repeat this for the remaining root joints.









06 Connect the FK to the IK Now we will be connecting the output rotation values for the IK joints to drive the _drive group node on our FK controls. Let's tackle this for the index finger. Select both indexD_fk_ctrl_drive and indexD_ik_jnt, and go to Windows>Node Editor. Open both nodes up fully and then connect the output Rotate attribute of indexD_ik_jnt into the input Rotate attribute for indexD_fk_ctrl. Do the same for indexC_fk_ctrl_drive and indexB_fk_ctrl_drive. Then translate index_ik_ctrl around to see if it drives the FK controls. If all works well, repeat the process for the remaining fingers.

07 Connect the fingers to the palm If we move the arm controls around, you'll find that the fingers will stay as they are. Before we fix this, let's clean up the Outliner. Select all the finger root joints and hit Cmd/Ctrl+G, and then rename this group 'fingers_rig_grp'. Then select all the IK control _offset nodes and hit Cmd/Ctrl+G. Rename this group 'fingers_Ik_ctrl_grp'. With nothing selected now, hit Cmd/Ctrl+G twice and rename the top node 'fingers_ctr_grp_offset' and the group below 'fingers_ctrl_grp'. Select the _offset node and parent it under wrist_bind_jnt, zero out the Translate and Rotate channels to position the _offset group, and then hit Shift+P to unparent the _offset node. Lastly, take all the FK finger control _offset nodes and parent them under fingers_ctrl_grp. Next, select in this order: wrist_bind_jnt, fingers_ctrl_grp_offset and go to Constrain>Parent.

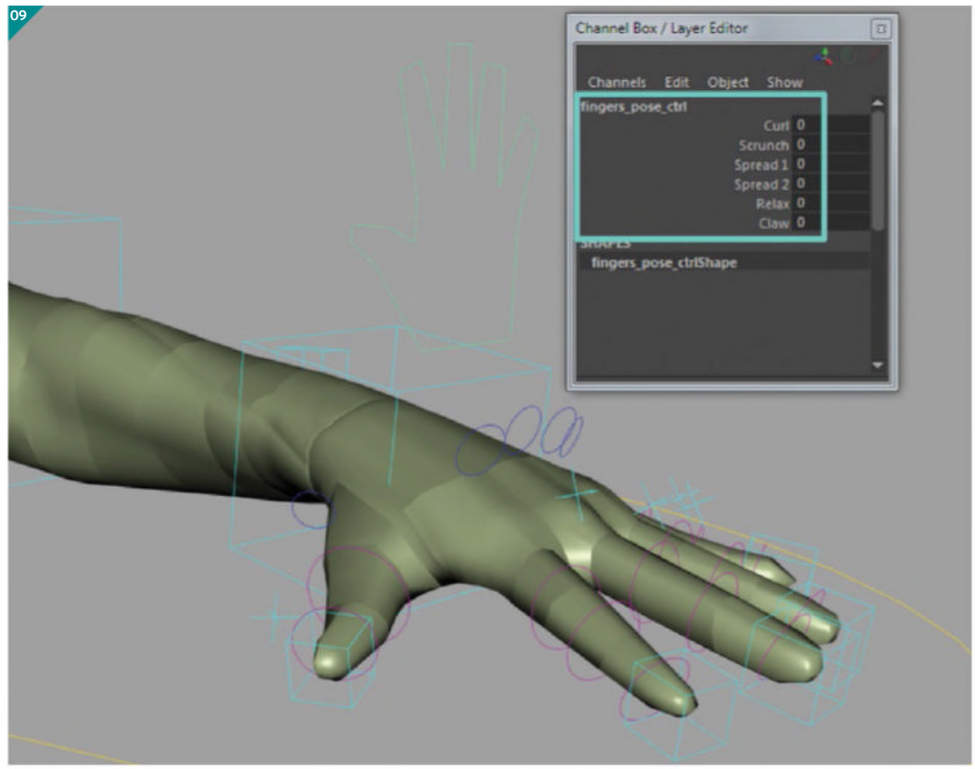
08 IK to follow or not follow We now want to create an attribute that will enable us to control whether the IK controls should or should not follow the rest of the rig. Before we do that though, and to make the tackling of this a speedier process, take the _offset node of each IK twist control and parent it under the relevant IK finger control. Now with nothing selected, hit Cmd/Ctrl+G twice to create two empty group nodes in a small hierarchy. Rename the top group 'fingers_ik_ctrl_grp_offset', and the group below 'fingers_ik_ctrl_grp'. Now select the _offset group, parent it to wrist_bind_jnt, zero out all the Translate and Rotate channels, and hit Shift+P to unparent it. Take all the _offset nodes for the IK finger controls, and parent them under fingers_ik_ctrl_grp. Select in this order: wrist_bind_jnt, fingers_ik_ctrl_grp_offset, and go to Constrain>Parent. Next, select fkIk_vis_switch_ctrl and go to Modify>Add Attribute. Create an attribute called 'ikFingers_follow' and give it a Min of 0 and a Max of 1. Now open up the Set Driven Key window and load fkIk_vis_switch_ctrl as the Driver object, and the parentConstraint node living under fingers_ik_ctrl_grp_offset as the Driven object. Make sure the IK Fingers Follow attribute is set to 0, check the Wrist Bind Jnt W0 (on the parent Constraint node) is set to 0, and hit Key on the SDK window. Then set both attributes to 1 and hit Key once more.

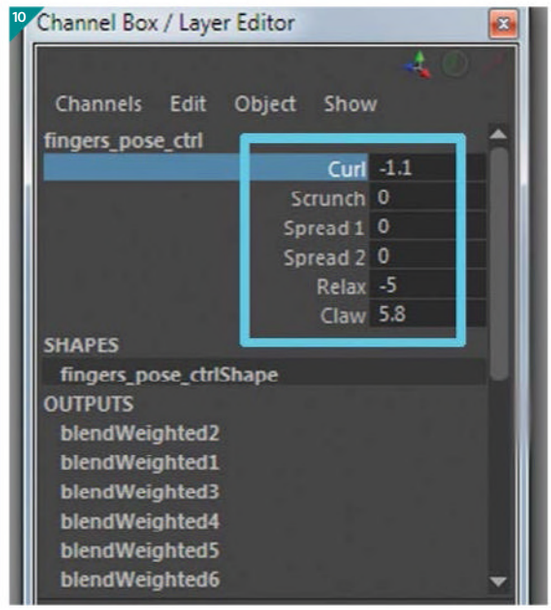
09 Finger pose control With the fingers now following the rest of the rig, let's look at creating some controls to allow the animator to pose the fingers quickly. For the control, use the CV Curve Tool and create a shape that mimics a hand's silhouette. Rename this control 'fingers_ctrl'. Translate and orient the control so that it sits close to the fingers, and scale it up if necessary. Once you are

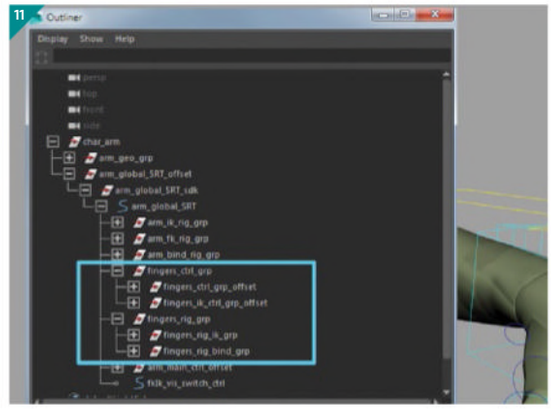
happy, parent it under fingers_ctrl_grp, and then with fingers_ctrl still selected, go to Modify>Freeze Transformations. In the Channel Box, highlight Translate, Rotate and Scale, and 'Lock and Hide' them using RMB. We now want to add some custom attributes that the animator can use to pose the fingers. With fingers_ctrl selected, go to Modify>Add Attribute and create the following: curl, scrunch, spread1, spread2, relax and claw, with the following settings: Data Type: Float, Minimum: -10, Maximum: 10 and Default: 0. You can also add additional attributes to drive each finger individually.

10 Drive the poses with SDKs With the attributes ready, select all the finger controls (other than <finger>A_ctrl), and press Up twice. This will let you pick-walk up to the _sdk level of each control (which is what we will use to drive the finger attributes). Alternatively, go into Outliner and select all the _sdk groups for the finger controls. With the _sdk groups selected, go to Create>Quick Select Sets, and call it 'finger_sdk_set'. This will let us come back and select the _sdk nodes if need be. With the _sdk nodes still selected, go to Key>Set Driven Key>Set. This should bring the selected groups into the Driven window. Select fingers_ctrl and hit Load Driver. The entire rig should be in default position and the finger_ctrl attributes should be at 0. Now, highlight Curl in the top-right box, all the _sdk_groups in the bottom-left box, Rotate Z in the bottom-right box and hit the Key button. Set the value of Curl to 10, use only the Rotate Z channel of the _sdk group nodes to curl the fingers and create a fist, and hit Key once more. Going between 0 and 10 now with the Curl attribute should curl the fingers. Set Curl to -10, select all the _sdk group nodes, and use Rotate Z to go in the opposite direction and hit Key once more. Let's move on to Scrunch next. In the top-right box of the SDK window, highlight Scrunch, and with the same settings (all the _sdk groups and Rotate Z highlighted), hit Key when everything is at 0. Then set the value of Scrunch to 10, and use only the Rotate Z channel to create a pose of someone scraping their fingers against a blackboard. Hit Key when you are happy, and do the opposite at -10. Using this method, create poses for the remaining attributes on fingers_ctrl. You'll find that sometimes, you'll need to use a different Rotate attribute. For example, for the Spread attributes we have used Rotate Y.

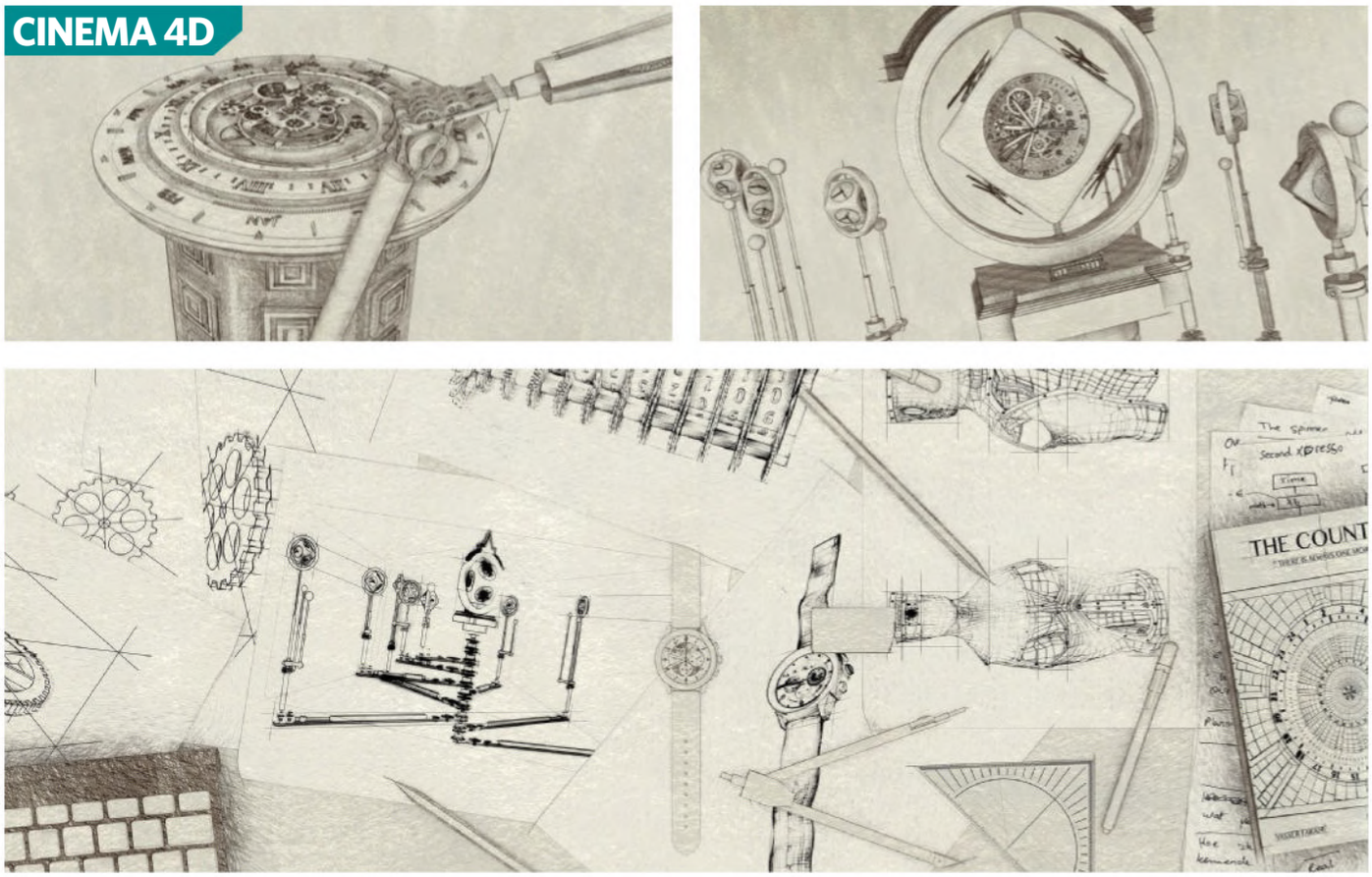
11 Finalise the rig Start by adding a new attribute to fkIk_vis_switch_ctrl and then call it 'ikFingers_vis'. Give it a Min of 0 and a Max of 1. Then use Set Driven Keys and use this new attribute to drive the Visibility on fingers_ik_ctrl_grp. This should help keep the viewport a little cleaner. In the Outliner now, select all the finger control groups and hit Cmd/Ctrl+G. Rename this group 'fingers_ctrl_grp'. Then take all the groups holding the finger joints, group them together and call this group 'fingers_rig_grp'. Take both of these groups and parent them under global_SRT_ctrl. Next, create one more control that will let you translate the entire arm in either FK or IK mode. In the Top view, just use the CV Curve Tool to create an arrow shape (or a shape of your choice). Then name this shape 'arm_main_ctrl' and hit Cmd/Ctrl+G twice to create two group nodes above the new control. Rename the two group nodes appropriately with the _sdk and _offset suffixes. Now take the _offset node, hold down the V key on the keyboard (for point snapping) and snap the control to wrist_bind_jnt. If you cannot select the control easily as it is penetrating the mesh, edit its shape in Component mode. Parent the following groups under arm_main_ctrl: arm_fk_ctrl_grp and arm_ik_grp. The latter group should be housing elbow_ik_ctrl_offset, arm_ik_ctrl_offset and arm_root_ik_ctrl_offset. Then take arm_main_ctrl_offset and parent it under global_SRT_ctrl. Test the arm to see if it all works correctly, and if so, go in and pop all the relevant parts into the relevant Display Layers. 'Lock and Hide' any attributes you do not want the animator to key, and hide anything important (such as the IK handles) to finish the rig. The rig should now be complete and you can now skin the geometry to the joints.







CINEMA 4D

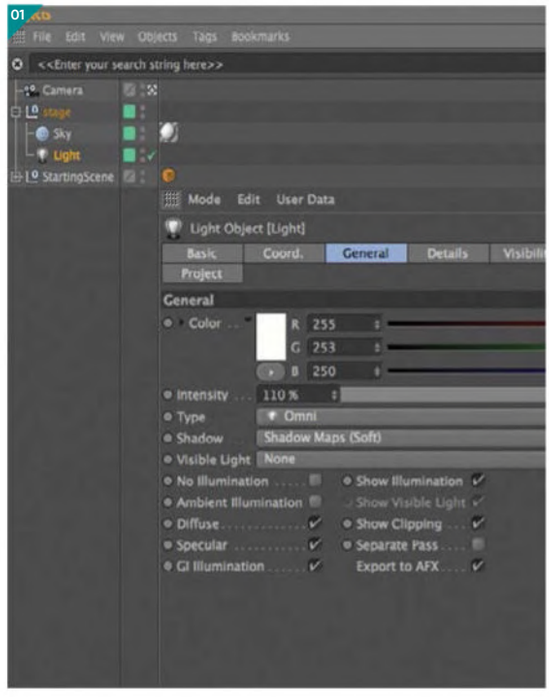


Hand-drawn style animation

Cinema 4D
Yasser Farahi
www.ovalpicture.com

Yasser Farahi is a motion-graphic and concept designer based in the Netherlands. He has a bachelor of design degree from Utrecht School of the Arts.

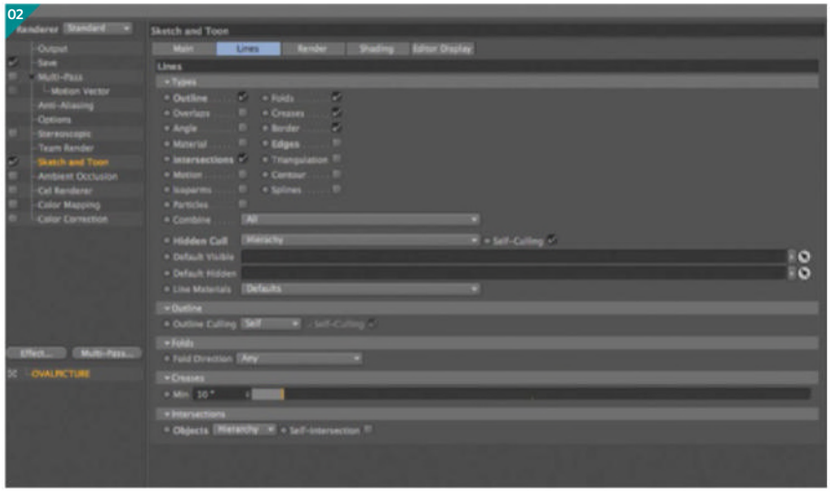
The Sketch And Toon function in Cinema 4D enables you to give your 3D animations a simple, but incredibly complex-looking hand-drawn or painted look. The fun thing about this tool is that it will enable you to create your own signature style, helping you to become the kind of artist that you've always wanted to be. In this tutorial we will take a look at the basic workflow of this powerful tool. Over the next few steps you will learn to create Sketch materials, and how to set up a very simple scene.

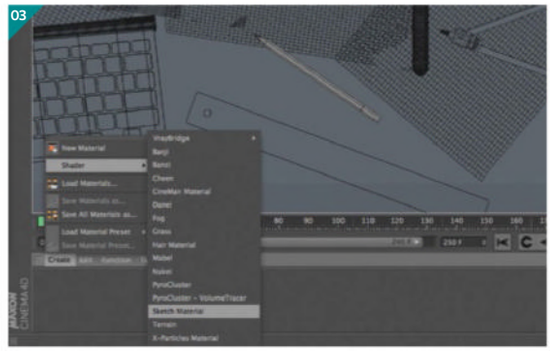


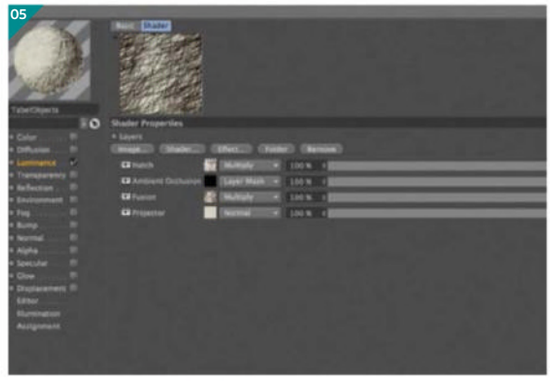
01 Lay out the scene Open the start scene files that can be downloaded to accompany the tutorial. Create a Null object and rename it as Stage. Add an Omni light and move it to the centre of the scene. Once it's there, increase its Intensity up to 110%.

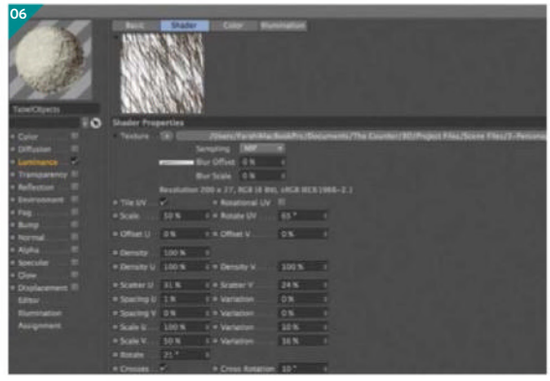
The next step is to create a Sky object, then select both the Sky object and the light source, dragging them both into the Stage (our Null object). Once you've done this, create a new material, turning off its Specular, then change its name to Sky Material. Go to the Color channel and click on the arrow in front of the texture, hovering your cursor over Effects until you can choose Ambient Occlusion. Once you've opened the Ambient Occlusion settings, change its Accuracy value to 100% and Maximum Samples to 100.

YOUR FREE DOWNLOADS
from **filesilo.co.uk/bks-798**
- Tutorial screenshots
- Video tutorial
- 'The Counter' animation
- Scene and texture files









02 Set up the render The next stage of the tutorial is to open up the Render dialog box in Cinema 4D. Here, ensure that you add Sketch And Toon to your renderer. Under the Lines tab, turn on Outline and Intersection, then change the Hidden Cull to Hierachy. For the last part of this step, go to the Render tab and choose Normal for Line AA Option. Now you should be able to close your Render Settings box and continue with the animation.

03 Add the sketch lines This is definitely the most important part of the tutorial, as now we will be defining the sketch lines for Cinema 4D. Let's create a new Sketch material by going to the Material section. Once there, click on Create, hover you cursor over Shader and select Sketch Material. Apply the material you have just created to the object of your scene. Double-click on this material and open the Material Editor. Rename this material to Sketch Lines or Drawing Lines. In the next step we are going to adjust several of the settings of this material, including the Stroke, Adjustment, Distort, Color, Thickness, Opacity and Clone.

04 Define the drawing lines The Stroke option will define the line surrounding your scene objects. Press Enable under Stroke, change Match type to Flat, change the Join Limit value to 50%, and Join type to Miter with a Limit value of 40. Under the Adjustment channel, turn on Relative under Overshoot option, use 3% for both Start and End values and use 50% for their variations. Distort will give the sketch lines a more natural look by distorting them. Turn on Curve Stroke and change its type to Akima, set the Step value to 5 and Strength to 14%. Change the Mode to Sin and use 5 as the Displace value. The Color channel will define your pencil or brush stroke colour, so opt for any colour of your choice, while thickness represents the density of your drawing lines. Turn on Along Stroke and change its mode to Spline. The Opacity channel will influence the transparency of your drawing lines. Let's say you want your artwork to look more like water paint – this option will help you achieve that. Last of all, the Clone channel will automatically clone more sketch lines around your objects, but be very careful with this option if you want to save yourself some render time.

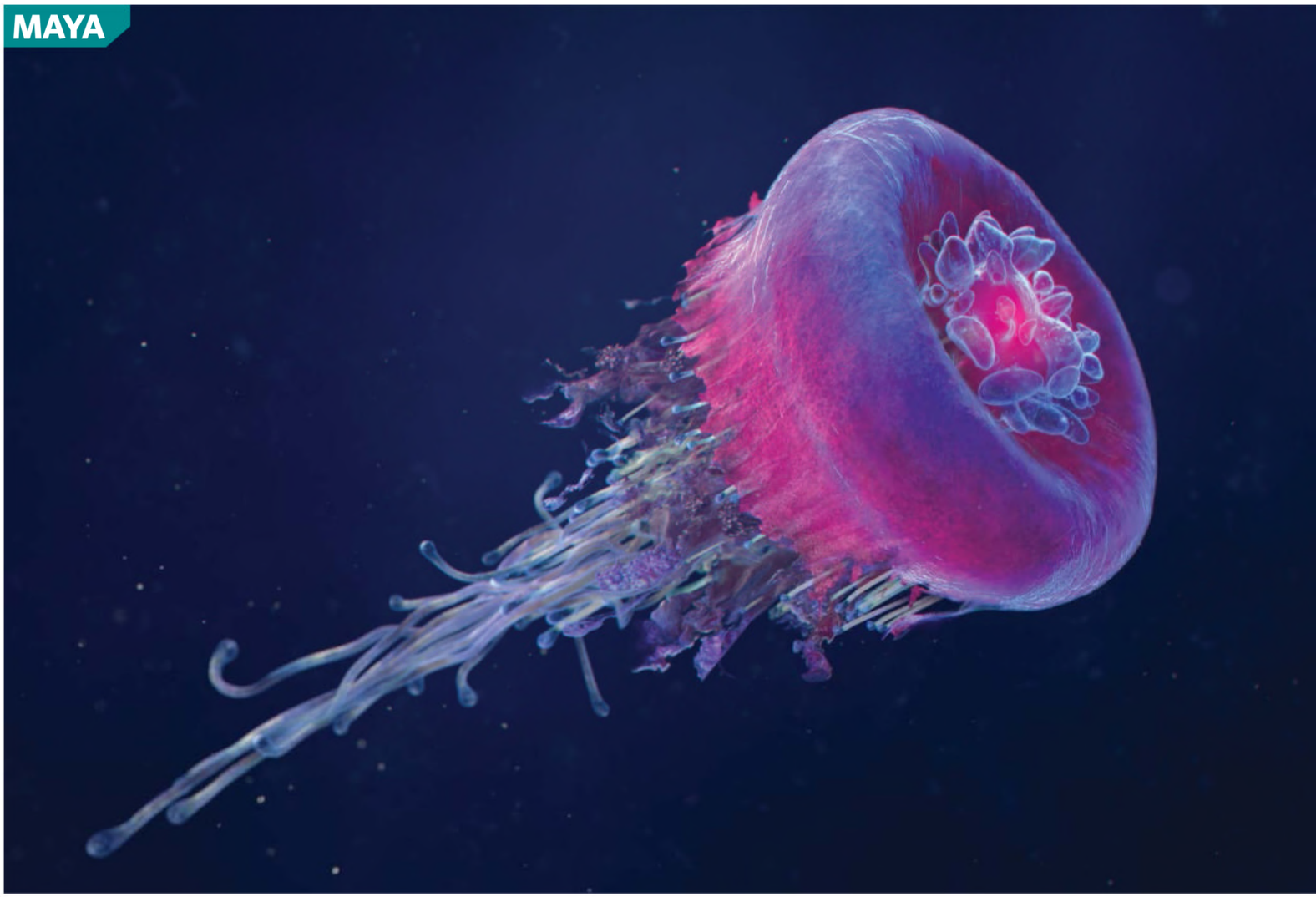
05 Create a Hatch material Create a new material and turn off Color and Specular. Turn on Luminance, then go to the Luminance channel and add Layer to the Texture channel. Open this channel and start adding a Hatch Material (Shader> Sketch>Hatch), a Fusion channel and a Projector (Shader>Effects>Projector). Copy the Ambient Occlusion channel from the Sky material and paste it right underneath Hatch. Next, click on Ambient Occlusion and reverse its colour order from black to white, to white to black. Go back to the Layer channel and change the order to Hatch (Multiply), Ambient Occlusion (Layer Mask), Fusion (Multiply), Projector (Normal). Open up the Projector channel's settings and load PaperTexture.jpg onto Texture, making sure that the Projection mode is set to Frontal.

06 Finalise the hatch Go back to the Layer channel, click on Hatch, load Brush.jpg onto the Texture channel (that represents your pencil) and adjust its values. Under the Color tab you can define your brush colour. Go back to the Layer channel, copy the Hatch material that you just created and paste it into the Blend channel and Base channel of the Fusion layer. Set the Fusion mode to Multiply and at 85%. Go to each one of these new Hatch materials, and under the Illumination tab, turn on the Light and Shadow option in both of the channels. Close the Material Editor and apply the materials to your scene objects.

Create Sketch materials

As one of the download files, you'll see that a final C4D file is included. Explore this file in your own time, playing with the settings to see the kind of looks you can achieve. Experiment with the Sketch And Toon tool to create your own signature look. You'll also find an accompanying video guide to help you.

MAYA



Animate a jellyfish

Maya, MARI, Photoshop

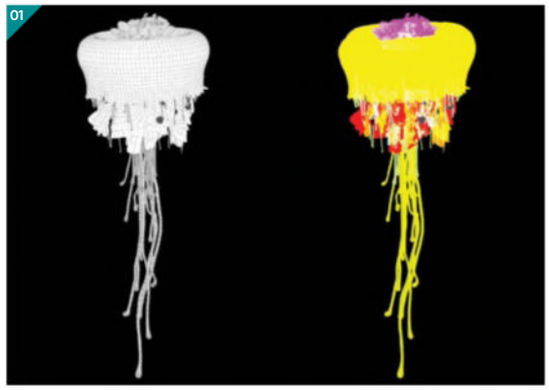
Valentina Rosselli
vrosselliportfolio.wix.com/vrossportfolio

Valentina is a highly motivated 3D artist with a background as a painter. She works in London as a freelancer

This tutorial will help you understand the pipeline process of building your own jellyfish from scratch and simulating it in an underwater environment with Maya. It will provide you the techniques to learn quickly and give your project life. We will discuss the best way to think about the jellyfish in relation to the animation, model according to the simulation needs, learn how to work with a Nucleus system (using nCloth, nConstraints and Wrap Deformers) and finally how to bake everything.

This tutorial is suitable for people who have a basic knowledge about Maya nSystem but also for people who may not and want to start using it. It is really important to be organised from the beginning. After you choose your preferred jellyfish, make sure you build a detailed concept according to the animation as explained in Step 1.

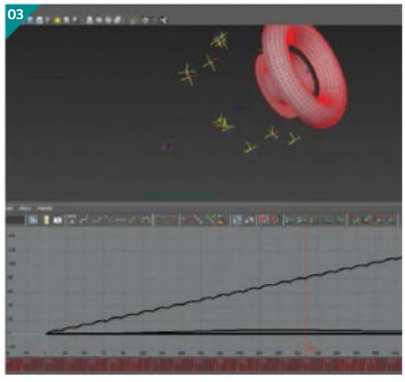
The best way to create it is to break the jellyfish into different parts thinking about it as if it is as composed of layers. For example, as you can see from the images of the tutorial, this jellyfish has one layer for the swimming bell, one for the shape on the top, three for the lappets placed inside the bell, one for the small structures on the lappet and several tentacles layers different in terms of scale, and each of these have been duplicated a certain number of times. It is necessary to collect video references to completely understand the way that a jellyfish moves. Make sure you test the simulation of each nCloth layer separately and once you are satisfied with the result, you will be able to build the master scene.



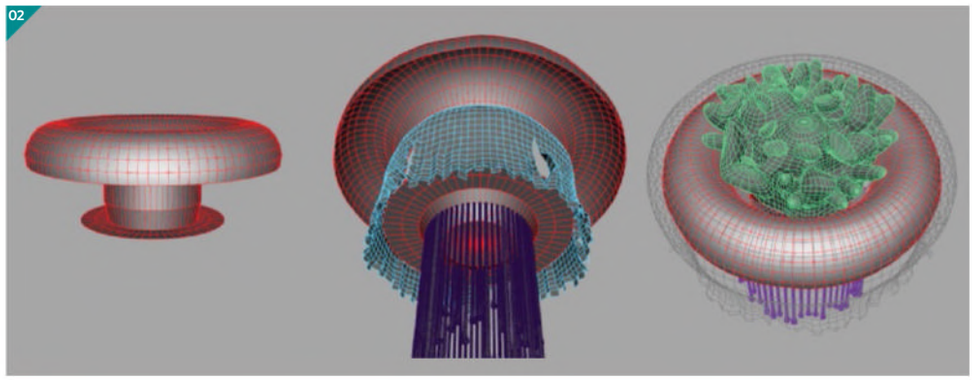
YOUR FREE DOWNLOADS from **filesilo.co.uk/bks-798**
• Tutorial screenshots

01 Model according to the animation Paying special attention to the jellyfish's structure, we can divide it into different layers in relation to its major surfaces and axes. As briefly mentioned earlier, we have the following layers: one swimming bell, one smaller bell structure on the top, three lappets, one for the small structures on the lappet and six types of tentacles, with each one duplicated a certain number of times. Layers have to be modelled in a straight pose, paying attention to not go too far with the number of polygons. As you can see from the images, meshes are combinable so once you have modelled the main structures you can also play with their position and scale. In the meantime make sure to design a proper supporting structure as explained next.

02 The supporting structure Shown as the red wireframe, the supporting structure is a Passive Collider mesh built to drag behind each nCloth layer and their nConstraints. In order to do that you have to link the vertices of the last edge loop of a layer with the one corresponding to the supporting structure. The supporting shape has to be designed in order to not collide with layers in straight pose. For instance, the swimming bell is modelled with a hole in the top in order to place its last edge loop next to the edge loop of the supporting structure. Go ahead with this method, taking into consideration the layers' position and move to Step 3 when you are satisfied with the result.



03 Start to animate Once you have the final model of the supporting structure let's move to its animation. This step is essential: you can choose the method that suits you better using an expression or key framing. Make sure to collect video references to understand the movements before animating it. The most important thing to know is that the structure will lead the animation of the nCloth. Once you have animated the supporting structure, make it a Passive Collider and go to frame one. The next step is to make each layer an nCloth.



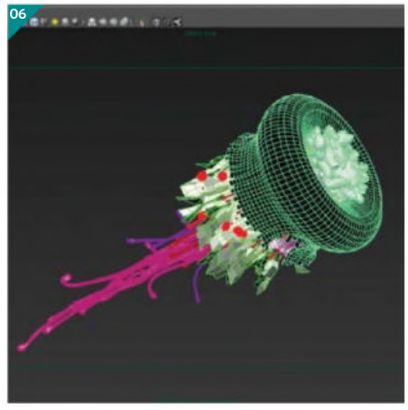
04 The underwater environment Once you've converted the layers in nCloth you need to deal with the settings of the Nucleus system. Its attributes depend on the nCloth properties and vice versa. To help simulate movements underwater it will be better to increase the air density and reduce the gravity in the Nucleus node. In order to add detail and realism you could create an emitter to simulate some particles underwater. In case you decide to add them you can play with Fields affecting the nParticles. Depending on the effect you want you can also add the Field to the nCloth layers of the jellyfish at the end but try not to go too far with them.

05 Add nCloth layers Each nCloth layer has to be connected to the supporting structure with the edge loops of nConstraints. Proceed layer after layer and go by degrees. Test the properties of each nCloth separately before building the master scene and keep in mind that each part reacts differently underwater. You also need to collide each layer with each other but be careful with heavy simulation. According to the settings of Nucleus, don't increase the mass of the nCloth too much. Also set Drag, Lift and Damp to a low value.

06 Bake simulation and tips To avoid having a heavy scene you could bake a simulation of the layer's vertices or build an nCache once you have completed the simulation. You could also build low-poly meshes for the layers just for the simulation and then assign them to corresponding high-poly ones with a Wrap Deformer. Simulating the jellyfish is obviously not the only method you can use to animate it, you can also do it with nHair and Joins.

Make sure your project is well organised

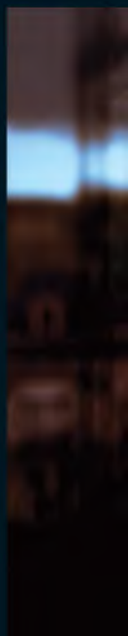
To avoid any issues during the animation process and to not lose time you need to design your concept at the very beginning of the project. Always test your simulation in separate scenes from the master one. The complexity of the simulation could give you a hard time so to avoid stress just be precise and organised with your project, it will be more fun doing it and you will hit the mark sooner.

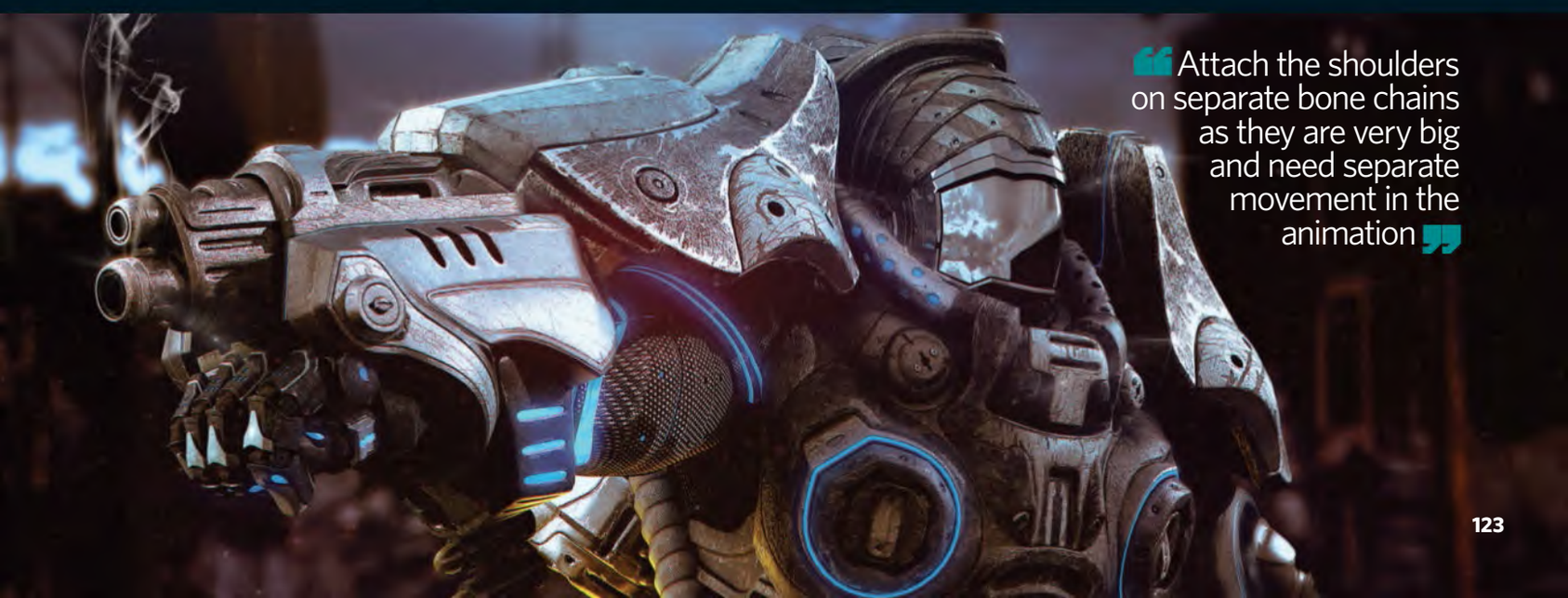


Animate a robot battle scene

Rig, light and render fighting robots fit for a movie

The development process of creating a finished artwork is important, so this tutorial will cover the main steps of conceiving an idea and then finessing it right through to the final image. It will teach you to prepare the characters for a post-apocalyptic scene in Maya, use a traditional rig and HumanIK rig to connect with motion capture files, create the materials with textures, create the light rig and use some V-Ray and Maya techniques. There will be different passes rendered including: Color, Specular, Reflection, Z-Depth, Render ID and Material ID. These elements will be assembled in Digital Fusion in the final composition to add some life to our characters. At the end we can play with different colour corrections to achieve the final look.





> Attach the shoulders on separate bone chains as they are very big and need separate movement in the animation

ILIYA ATANASOV
www.pixelhunters.com

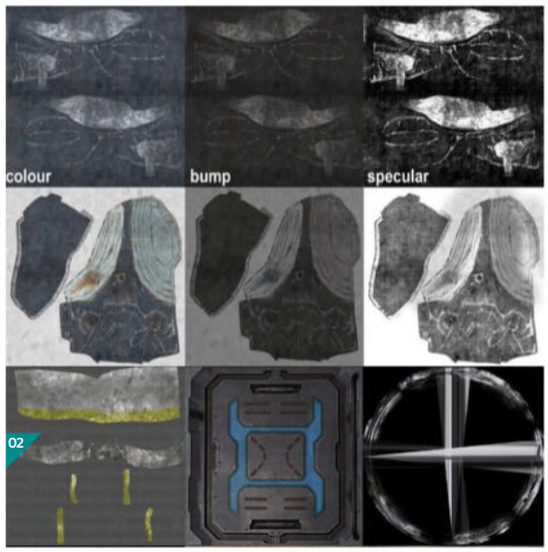
Software
ZBrush, Maya, V-Ray, Digital Fusion

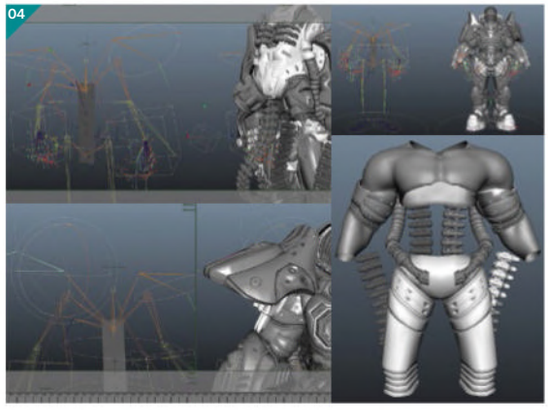
Learn how to
- Prepare a 3D character mesh for work
- Rig a character with a traditional Maya HumanIK system
- Choose composition, pose and prepare the rest of the elements by modelling smaller background city elements
- Create some smoke effects by using the Phoenix plugin for Maya
- Prepare the lighting and rendering for the scene
- Composite separate passes

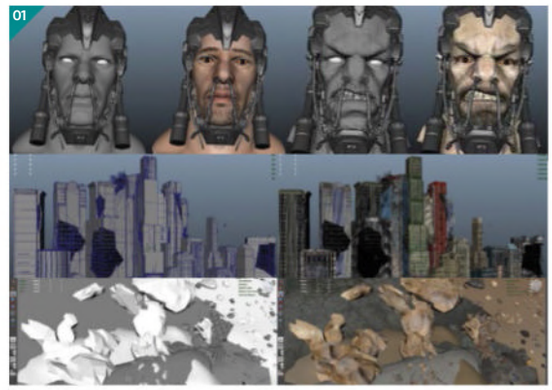
Concept
For the environment of this particular scene, I wanted to create a postapocalyptic world, so I found multiple references of ruined world cities. For the character fights I used references from games like *Tekken* and *Street Fighter*.

YOUR FREE DOWNLOADS from **filesilo.co.uk/bks-798**
- Tutorial screenshots

01 Add the face and background elements In the movie our character has two faces and changes from good to bad. In most of the scenes the face is hidden, as the helm is very reflective. The same modelling techniques can be used from ZBrush to model the faces. There will be a transformation shot between them at the end of the movie, so model the good/human face first and distort it later to achieve the bad/monster one from it. In this case the same number of polygons will be maintained and the morph will be possible between them. For the background scene, create very low poly objects of buildings, broken poles, fences, roads, multiple rocks, bricks and so on. Use simple techniques of extruding faces from cubes to achieve the results. Use the Cut Faces tool to add more spans for the objects and the Bevel and Crease tool for creating smooth, sharp edges.





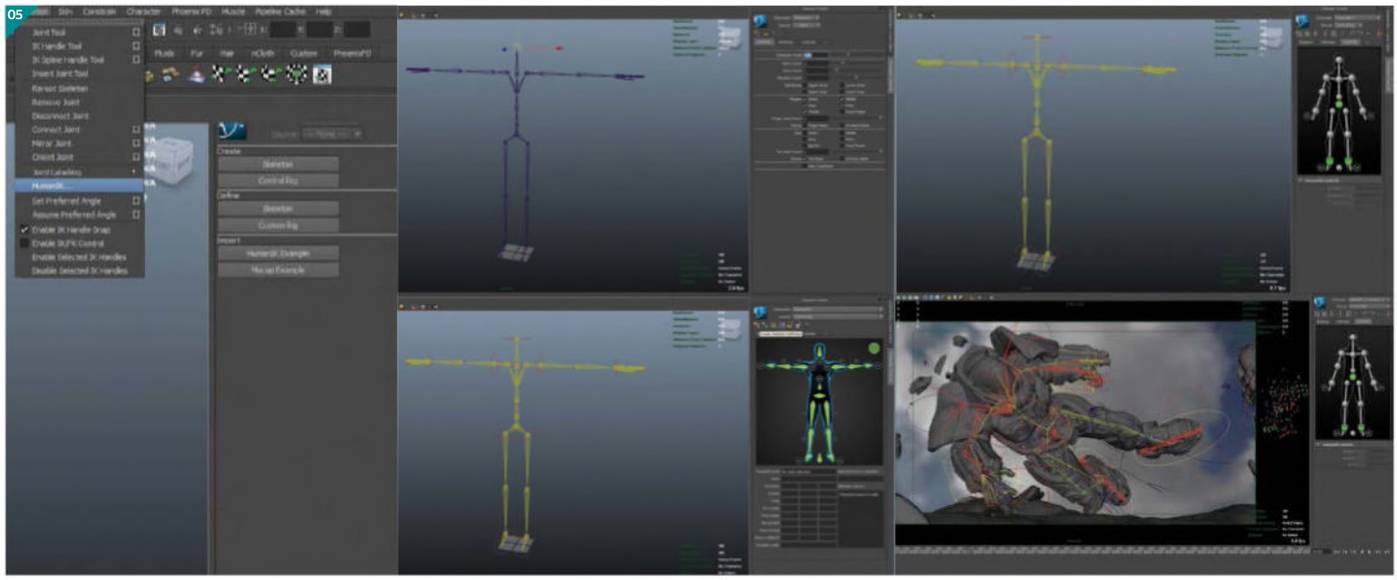


02 Texture the surfaces Every part needs to have unique UV coordinates so that it can be directly painted on top. Start with adding a fill colour texture, and after that create creases or scratches over the bended edges, following the UVs. Use different brushes to achieve different abrasive effects over the used metal. Desaturate the colours for an old-looking surface. Add dirt maps over the whole textures to create noise effect. Use different blend modes to add a couple of textures, one over another, for variety. For the colour maps, use Desaturate and Levels in Photoshop to create Bump, Reflection and Specular maps for the materials.

03 Smoke effects For the smoke use Phoenix software from Chaos Group. Create a fluid container, one source and a small sphere inside the container. On the container attributes, make Discharge 15 and Temperature 1,800 as you don't want fire in it. Run the simulation and save the cache. Reuse the same cached container for different locations inside the scene. Note that the simulation can be made in low resolution. Then use the resimulate option to add detail on top of the already cached simulation – this saves time when creating good-looking smoke effects. Don't forget to add wavelet output channels when simulating first pass. Then play with Time Scale inside Dynamics for slow motion effects.

Make the characters come alive!
The rigging and animation were made by Anton Gonzales and Rajesh RS. Use two different approaches for the character animation, a traditional hand-animated one and another driven by motion capture. First, create the initial rig for hand-made animation, you can parent or sync those controls later for the HumanIK rig.

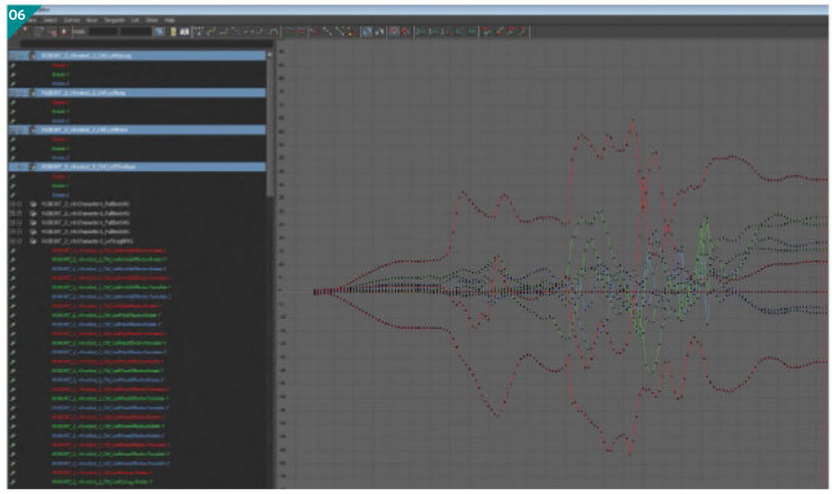
04 Rig and skin Make a custom rig, which includes a typical bone chain for every limb IK handles for reverse kinematics for the hands and legs. The upper body is one big piece of metal, so there is no bending in the spine area. This brings some limitations on the animation movement of the character. As the character looks robotic, the skinning process is very easy. Parent the different mesh parts to their bones. Only a couple of soft, bend elements needs to be skinned. Use smooth skin and paint the vertexes or mesh if necessary. These elements are the bullet chains, the costume on the hands, the trousers and the two pipes from the sides of the waist. On his back the character has two big engines, so add additional bone chains to move the heads of the engine. Do the same with the shoulders and attach them on separate bone chains.



Materials, lighting and rendering

The V-Ray renderer for Maya is used here. The shiny metal parts of the character armour will have a lot of blurred reflections, which will be easy to render. The overall mood of the movie is ominous as we are in a postapocalyptic world so we will use smooth shadows. Separate the background from the foreground with two types of lighting. This will secure faster renderings of the different passes. An HDRI image for the background and three-point lighting for the characters will work best.

05 Animation with HumanIK Open HumanIK and then create a Skeleton. Then match the size of the new skeleton with the size of the robot character using Character Scale. After that, choose Control Rig from the Source Falling menu. This will place the control rig for this skeleton. Now parent the new control rig points with the controls of the robot character – this means a traditionally made rig is now connected to the HumanIK rig. Then import the motion capture animation files so that they are inside your scene. Now create a skeleton definition and define every bone from the motion capture file to the HumanIK. This can be done by clicking periodically from every green bone from the right to the appropriate one (bone or locator) on the left. Lastly source or retarget the robot character to the defined motion capture character.

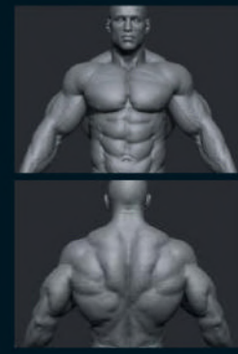


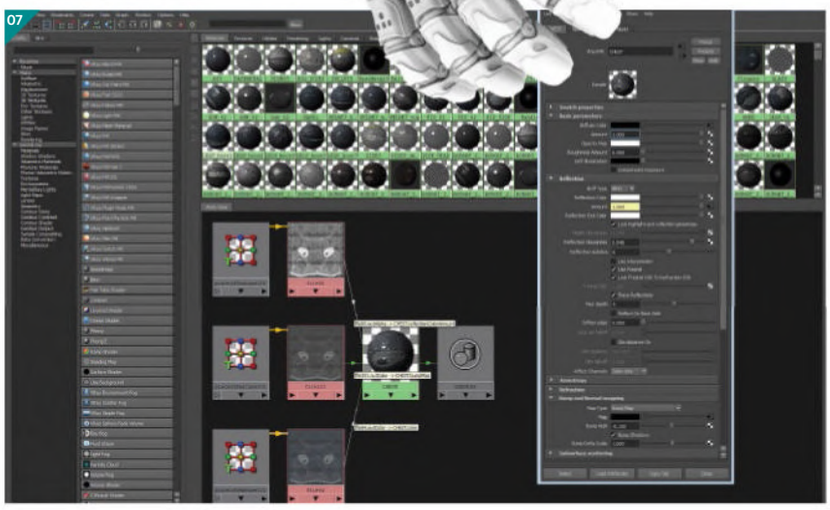
06 Shot planing with motion capture For the shot, we will use two characters fighting. Duplicate the same robot one more time, and change the materials later slightly. Make a rough layout for the timing and the action. In the rendered image both of them needs to be in the shot. Once we are happy, move to the motion capture section and clean the animation curves. Open Graph Editor and use the Euler filter to secure the gimbal's rotations on the bones. This action prevents the jumping of the rotation axes. You can also use the Resample and Simplify curve options to avoid nondesirable jumps on the bones when rendering with motion blur later. This problem is not visible with a standard playblast or with the changes in every frame, but when you go to subframes, you will get an issue with the motion blur.

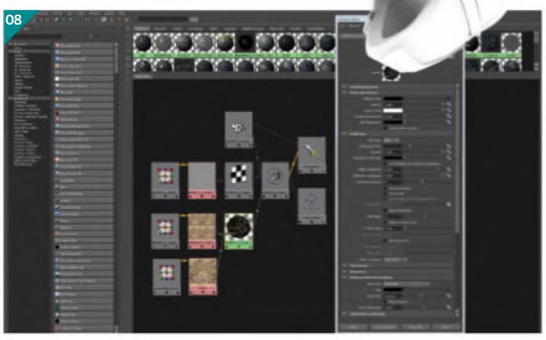
SIAMAK ROSHANI
siamakroshani.com

"I modelled and designed the character in Maya. Before modelling, I did some sketches in Photoshop. I used the Edge Extrude technique to model the character. Once I finished the model and UVs in Maya, I took it to ZBrush. For detailing in ZBrush I usually use surface noise and brushes such as ClayBuildup, Dam_Standard, Standard and Smooth Peaks. For texturing I used Photoshop and every part has its own texture (4k and 2k size). I used a very basic image for the texture base and 80 per cent of the textures were hand-painted in Photoshop. The character was done in 2010 (a personal project), and Pixelhunters asked to use it in their cinematic. The head model and texture inside the helmet were done by a different artist."

Showcase

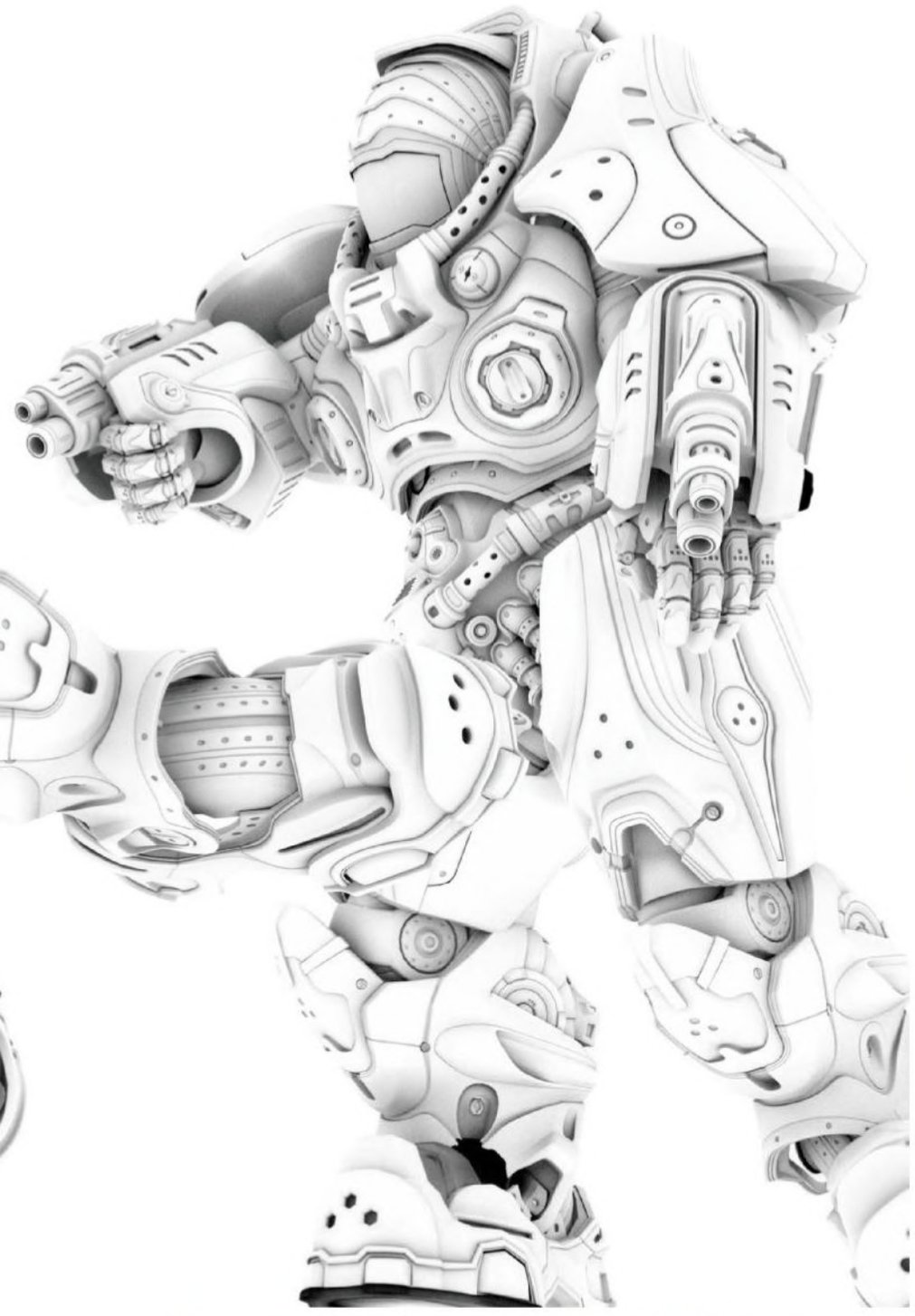






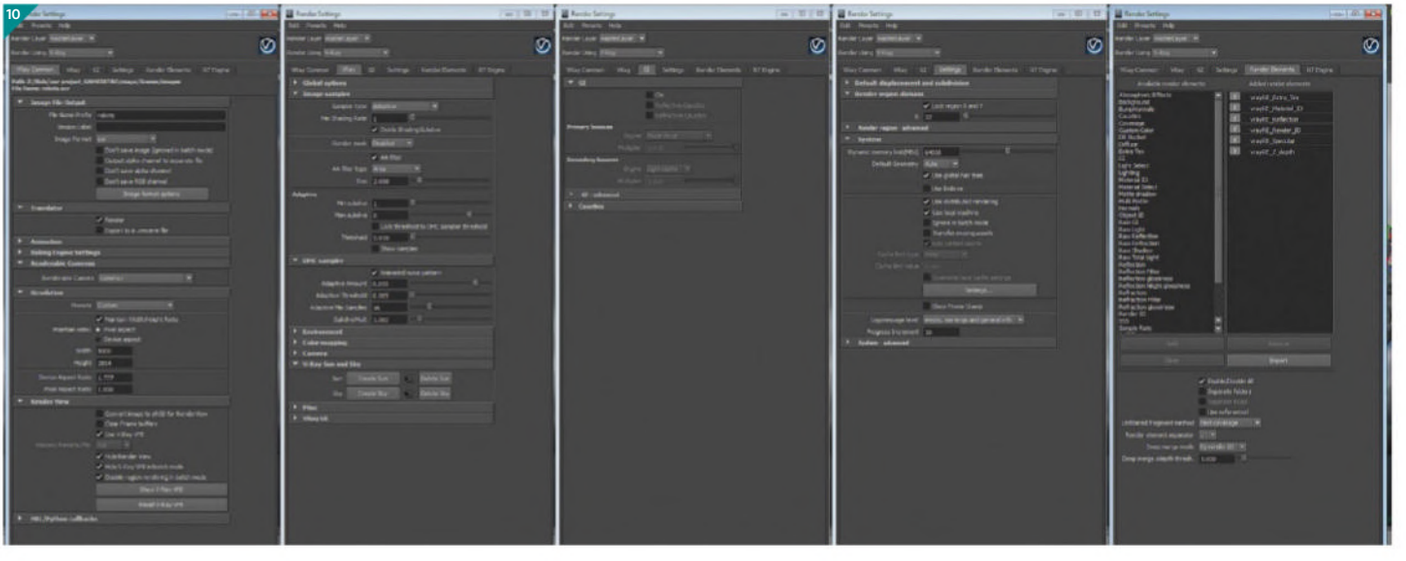
07 Texture the metal Now use a Standard V-Ray material. With the already prepared maps for Color, Specular, Reflection, Bump – connect them to the appropriate slots. For the very reflective parts like the chrome you can use Reflection Glossiness on 1 and control the Reflection Color amount for your own desired results. Some parts like the chest armour is not so reflective, so blur the reflections when you reduce the Glossiness factor. Do not forget that the number of subdivisions needs to be raised up, otherwise on the close-ups the noise will be too visible. Some materials like the side pipes and the engine need to have some highlight spots, so uncheck the 'Lock Highlight and reflection glossiness' option so that you can control the diameter of the highlights separately. On every shot the Bump multiplier must change and this will all depend on the distance from the camera. Use the same materials for the two characters and only desaturate their textures and change the glowing VrayLight material from red to blue for all of the incandescent parts.

08 Texture the ground and background Use the same type of V-Ray materials for the ground, rocks, buildings and so on. For the ground material let's use Displacement for extra details. Draw 32-bit EXR files either from ZBrush, Mudbox or Photoshop. Depending on the program and the way it exports change the Alpha gain and Alpha offset to 2.2 and -1.1, where 'Alpha is luminance' is checked. For the rocks and buildings that are further away, because of the destruction you can raise the Bump levels from 35 to 50 in the multiplier. To assure some secularity on the surfaces, lower the Highlight Glossiness to 0.855 and do not forget to turn off the Trace Reflection parameter.



09 Lighting Create a VRayLightDome (only half dome), and attach it to some diffused sky image. This will be the main ambient light model and you can use it to light the background elements – the ground, poles, buildings and rocks. Add a couple of rectangular V-Ray lights to light the characters. The light colour needs to be slightly blue with an intensity of 20. Make all of them invisible, but visible in reflections. Two of them need to be lit from right to left, one needs to light the red character from below, as he is falling to the ground and will be too dark. A couple of spherical lights can be placed around some parts of the character's shoulders and around the waist area. For the red character make them red, for the other one – blue. Apply a strong intensity multiplier of 5 with a radius of 1. Keep in mind that all values are scale sensitive and that the reference for Maya is set in centimetres. Lastly you can add more detail in the shadows so add one directional light – a bluish colour with 1.1 intensity to light the scene from left to right.

10 Render Settings Open V-Ray Render Settings and choose the 32-bit EXR Image format. In the V-Ray tab raise the Max subdivs to 8 samples. Turn on DMC sampler to animate the noise pattern. The GI is not turned on here to save some long shot rendering time, and it suits the dark atmosphere too. Do not forget to work in a linear workflow with gamma 2.2. Try experimenting with the HSV Exponential type of the Color mapping section. Reduce the region divisions in the Settings tab if you intend to use multiple computers for rendering. Dynamic memory must be set up close to the RAM of your machine. Click on 'Use distributed rendering' if you have a render farm and enough licences. Now add render passes to be used in compositing. This image was rendered in 35 minutes with one computer using dual Xeon processors at 2.9GHz.



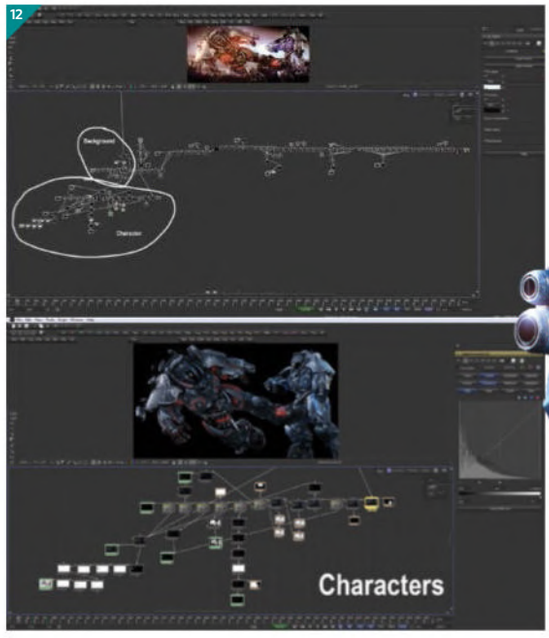
11 Use the essential passes There are a couple of important passes for assembling the composition. For the characters these are Beauty, Reflection, Specular, VRayDirt (Ambient Occlusion) pass, Z-Depth, Render ID and Material ID. The Material ID pass is the VRayLight material of the small glowing tubes flowing through his body. Use this pass to make these tube parts stronger with a small halo around it. Render ID is a pass which can stay as a backup, if you need some extra masks later on you can always chroma key it from this pass and use it in the composition. In general it attaches unique Material ID to every object in the scene and there is no need for any initial setup of the pass, just keep in mind that additional edge blur filters might be needed.

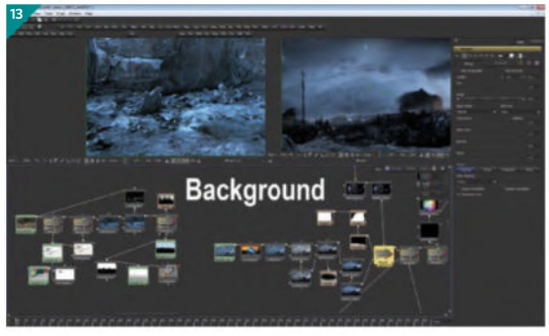
12 Composition of the characters One chain will be for the characters, another for the background. Start with the characters on top of the Beauty pass then add the Reflection pass onto the screen with a 0.6 blend. Then on Screen mode add the Specular pass. On Multiply mode add an AO pass with 0.8 value. For this render, only one material is used for both characters, so in the composition add a colour corrector on the Material ID pass with a polygon mask to separate them from one another. Add a soft glow and a normal glow on top and 'screen' it again over the other passes. Use Luma Keyer to extract highlights from the Specular and Diffuse pass to add on top of the shiny armour. As the overall lighting is set up to be very blue, the composition will look bluish here.

13 Composition of the background Apply a very similar process here as well. Another approach for the long shots is to render one frame of the background and composite it – in the same way as the characters – and after that just use the camera movement to render the offset position of the background plane. Of course there will be no parallax or perspective change when the camera moves, but for far away and blur backgrounds this will work. The render is then projected onto the plane and placed behind the characters, using the same movement from the 3D camera.

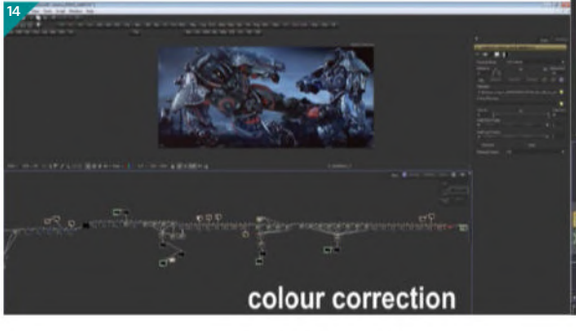
Compositing

This is the process needed to make the picture more beautiful via assembling all render parts. In this case, there is a separate background layer, foreground layer for the two characters, additional broken metal parts from the red character and multiple passes of dust and particles from the atmosphere. Use float 32-bit mode for the whole process with colour corrections and blending, and then save as an 8-bit single file. Some extra plugins may be handy like Sapphires and REvision Effects.





14 Finalise the colour correction Add a couple of defocus filters, lens flares, and some vignetting on the edges of the picture to achieve more of a film-look to it. Filters like distort RGB will put some extra RGB offset of the channels around the edges of the objects. Use the unsharp mask filter to exaggerate some elements of the characters like the hands and the head. Finally for this part of the movie the characters are in a kind of memory or dream, so the colour grading needs to be different from the original blue theme that we talked about earlier on in the tutorial. For that purpose add a couple of colour correctors and tint filters to offset the gamma into a more orange or red-ish tone. Work separately for all of the shadows, midtones and highlights. Now you can apply levels to crush the highlights and dark tones a little bit more, and this will secure a stronger contrast.



Showcase

Iliya Atanasov

I am a 37-year-old Bulgarian CG artist. My experience includes working for TV channels, advertisement agencies, film companies and game studios. I am the co-founder and studio director of Pixelhunters, which is based in Dubai, United Arab Emirates. I mainly focus my work on games and game cinematics.

Coliseum scene **Maya, Digital Fusion (2014)**
All of this was modelled in Maya, with some Despona models for the sculptures and columns. This was rendered with V-Ray.

Malaysia Dream **Maya, Vue Infinite, Digital Fusion (2012)**
Maya was used for the foreground and water simulations, Vue for the mountain environment and V-Ray for the rendering.

Water Huts **Maya, Digital Fusion (2011)**
Fast modelling for the houses, Maya was used for the paint effects and Xfrog for the trees. All rendered with mental ray.

144

140

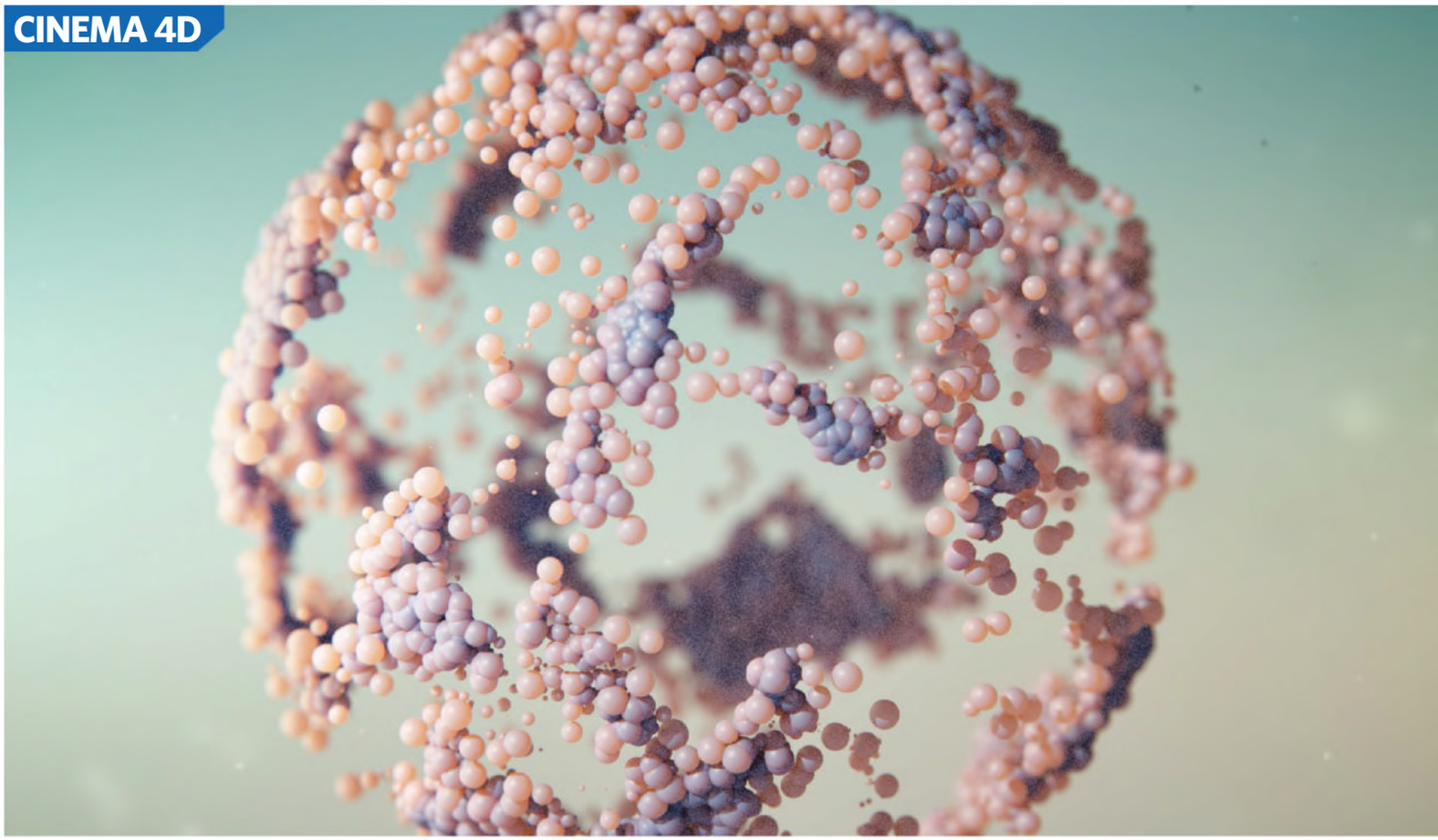
132

Visual Effects

136

146

CINEMA 4D



Create abstract particle renders

Cinema 4D, OctaneRender, X-Particles

Joey Camacho
rawandrendered.com

Freelance motion and graphic designer based in Canada, specialising in conceptual designs

Creating interesting images with particles and Cinema 4D is now easier than ever with the integrated plugin X-Particles, and creating abstract and organic imagery that looks and feels real can be an interesting challenge.

Creating abstract imagery is just as technical as creating real-world imagery. There are many things to consider, such as composition, colour, lighting, scale and camera settings. While you can jump in, tweak some settings and hit render, the beauty of creating compelling abstract imagery is in the details. If you make good decisions from the start, you'll arrive at a more effective result quickly.

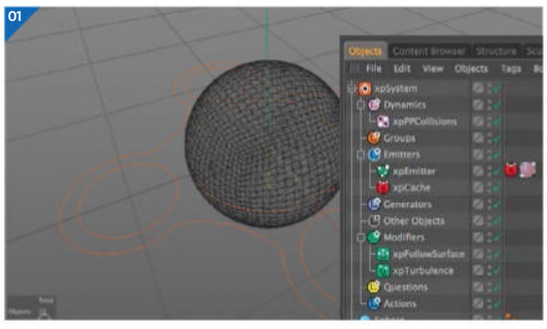
MAXON Cinema 4D is our 3D application of choice as it is more intuitive than other applications. With MAXON Cinema 4D Studio, the Mograph toolset is powerful and easy to dive into, and you can experiment with dynamic simulations that are extremely realistic.

X-Particles 3.0 is the most comprehensive particle simulation tool for Cinema 4D, and gives control over every aspect of the particle pipeline. While we have not dived into advanced features – like Python, or XPresso Nodes – we have enjoyed the ease of forming interesting simulations.

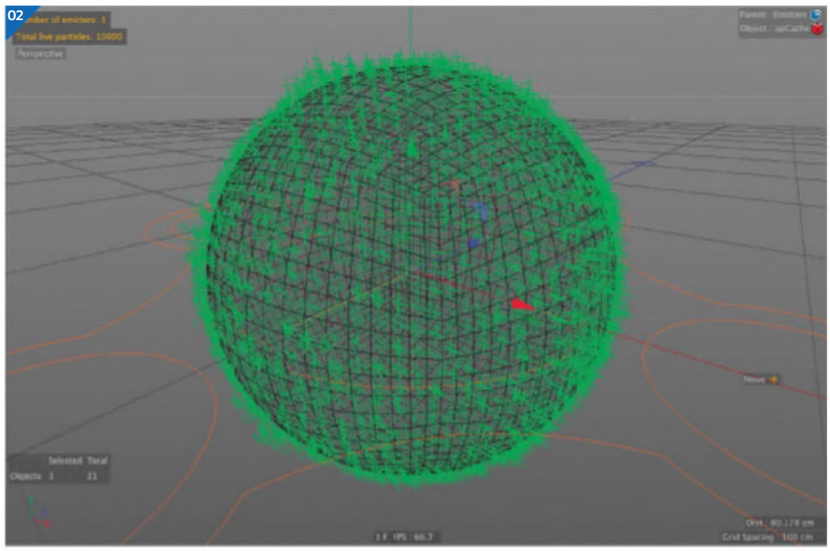
OctaneRender for Cinema 4D is our render engine of choice. It is the world's first and fastest GPU-accelerated, unbiased, physically correct renderer. Its development has been astounding, and it visualises renders nearly in real-time, which provides more time creating and less time waiting.

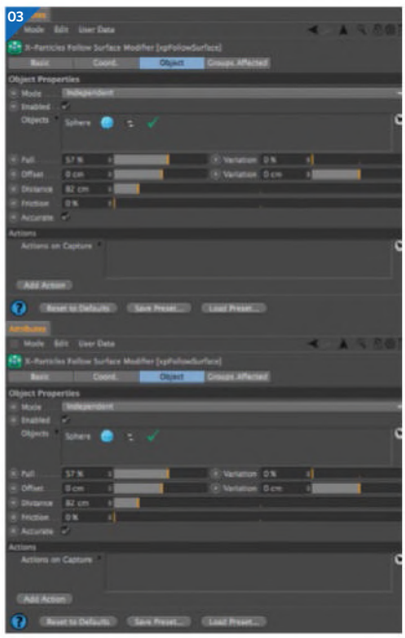
This tutorial will follow a traditional process, where we jump into the software with a rough idea, and see what comes up in a couple of hours.

Then, we'll show you some mixed lighting techniques and how to add intuitive Octane materials with subsurface scattering, which emphasise the organic feel of the image. In the end, we'll use Adobe Photoshop to add effects and imperfections to enhance the sense of realism.



YOUR FREE DOWNLOADS
from **filesilo.co.uk/bks-798**
- Tutorial screenshots
- Scene file
- Video tutorial





01 Set up the X-Particles system

First we'll set up a basic X-Particles system. This helps us to stay organised and keep the order of operations in Cinema 4D in its proper hierarchical order. By doing this we ensure that errors don't occur in the calculations of simulations. With the X-Particles system set up, you can visualise where emitters, generators, modifiers, and other objects should go in the object manager.

02 Emit from an object's surface

Next, we'll add an xpEmitter and a sphere. In the emitter's object tab we'll tell it to use the surface of the sphere as the source to emit particles from. Since you'll want to emit particles from the first frame only, go to the Emission tab, and select Shot from the Emission mode drop-down menu. Choose how many particles you'd like to be emitted in the Shot Count field. You can start with a lower number, say 5,000, to preview simulations quickly. Lastly, choose the radius of your particles, create some variation in their scale, and reduce the particles' speed to zero.

03 Add X-Particles modifiers

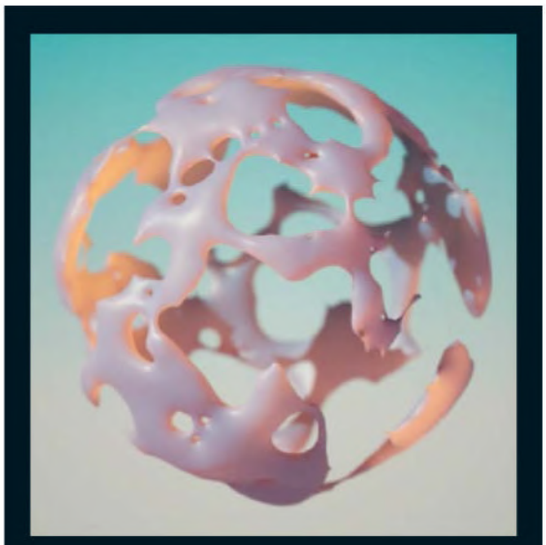
In this scene, we'll add two modifiers, xpTurbulence and xpFollowSurface. The xpTurbulence modifier moves the particles around in an organic manner with different variations. The xpFollowSurface modifier pulls moving particles across whatever object's surface it is referencing. In this example, we're going to add the same sphere to the xpFollowSurface object list. This way the objects will emit on the sphere's surface, get moved around by the turbulence, and will follow the geometry of the sphere.

04 Shake things up

X-Particles' turbulence modifier has many options, but its curl noise is very interesting. By adjusting the Blend and Add values, you can create interesting and organic movements. You can play with your simulation and adjust these values, as well as the scale, frequency, octaves and strength values to achieve different results.

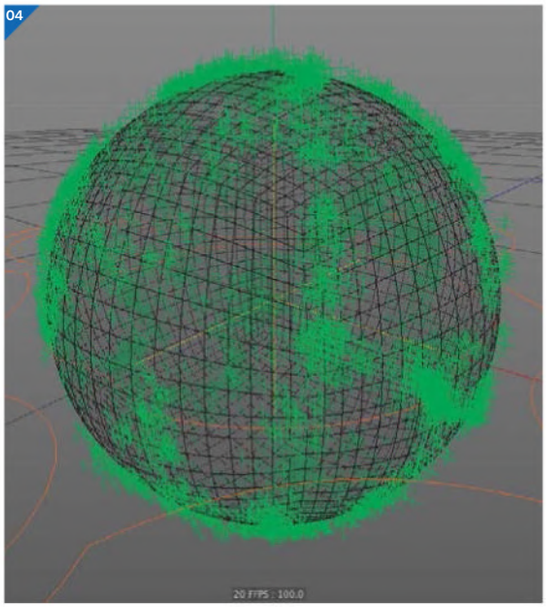
05 Add particle collisions

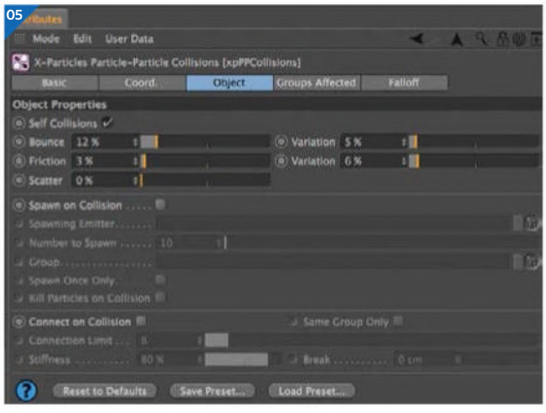
Up until now, our particles only interact with the surface of the sphere. They are not interacting with each other and are intersecting. One way to prevent this is to add an xpPPCollisions dynamics object and check Self Collisions. The particles will now interact with each other, bouncing and sliding away once they collide. Some will still intersect, but not as much as before. Keep in mind, adding this increases the amount of calculations that X-Particles needs to make, and the simulation will not run as fast. Not to worry though as we can cache our particle simulations.



Show some skin

The X-Particles Skinner object is another great tool to experiment with. You can quickly 'skin' to your particles to create organic forms or fluid like shapes. Basically the Skinner object creates a mesh from emitted particles, that you can customise with different parameters.



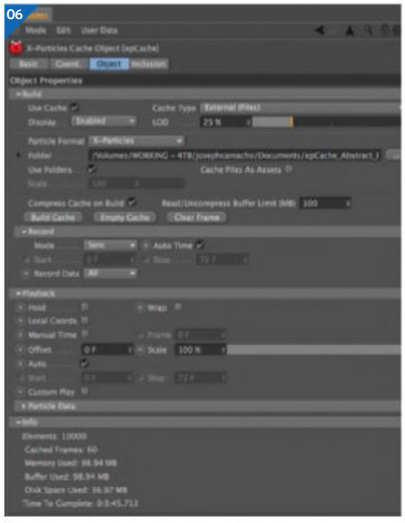


06 Cache the simulation The xpCache object is essential to saving, visualising, and scrubbing easily through heavy particle simulations. It also ensures that what you see in your viewport, or Octane's Live Viewer is what you will see in Cinema 4D's Picture Viewer and your final render. With that, add an xpCache object to your scene, and click Build Cache. X-Particles will save the simulation to your hard drive and you can scrub through to select a frame on your timeline where you like how the particles look.

07 Set up the Octane Camera Since this is an organic, almost molecular structure, you should give some consideration to the focal length and depth of field of your camera. Macro lenses are famous for their shallow depth of field, and often are 100mm or more. This emphasises the tiny scale we're trying to emulate. Set up an Octane Camera, and in the Attributes manager change the focal length to 200mm and experiment with other values to see the different effects. One of the most convenient features of OctaneRender is the ability to quickly select your focus distance by clicking on the object or particles you want to focus on. You can do this in the Live Viewer. Deselect Auto Focus in the thinLens tab of the Octane Camera tag, then increase the aperture value to create a more shallow depth of field.

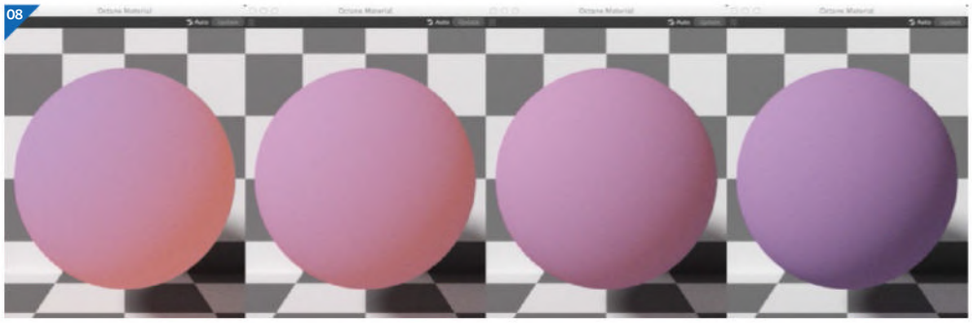
08 Subsurface scattering materials We chose to use a material with subsurface scattering because it can add a sense of life to objects. First, create a new Octane Diffuse material, and choose a base colour in the Diffuse tab. Next go to the Transmission tab, and add an Octane RGBSpectrum texture to the Texture channel. Choose the colour you would like to transmit once light goes through the particle. Finally, in the Medium tab, add a Scattering Medium texture to the texture channel. This will give you absorption and scattering textures. Most times, you can leave Absorption at 0 and Scattering at 0.5. The scale slider is what controls how easily light can be absorbed and transmitted through the object – or in this case, the particles. We've used 1.75.

09 Lighting Next add an Octane Daylight object to the scene. It comes with two tags, an Octane Daylight Tag, and Sun Expression Tag. First, make sure the Sun Expression Tag is enabled by checking it under Basic. Control the placement of the sun by using different longitude and latitude values, and experiment with different looks (sunrise, midday and sunset). Now select the Octane Daylight tag. The options in here are intuitive and flexible. You can increase turbidity, power, north offset and sun size, so experiment again. We've changed the sky colour and sun colour to be warmer and brighter.



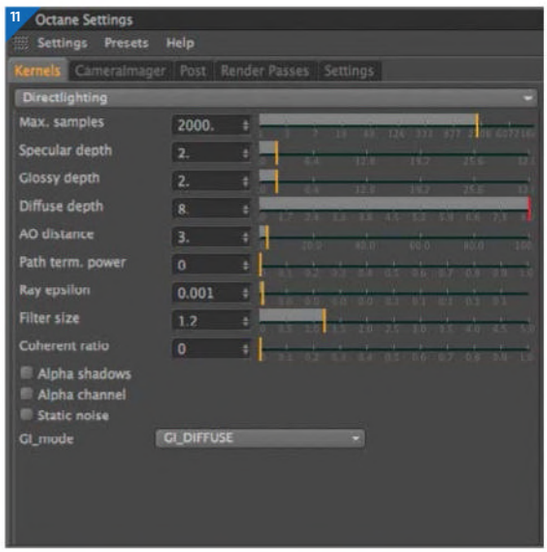
Start, experiment, finish

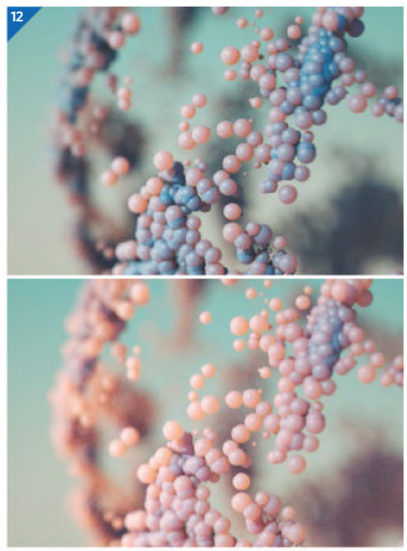
X-Particles has many different modifiers and parameters that you can manipulate to create amazing images. Take the time to try out different tools in different combinations to see how they work. Once you sit down and play around, the hours will fly by and you might not have something completed. Try to give yourself a set time to experiment and come up with a completed render to share with the Cinema 4D community.



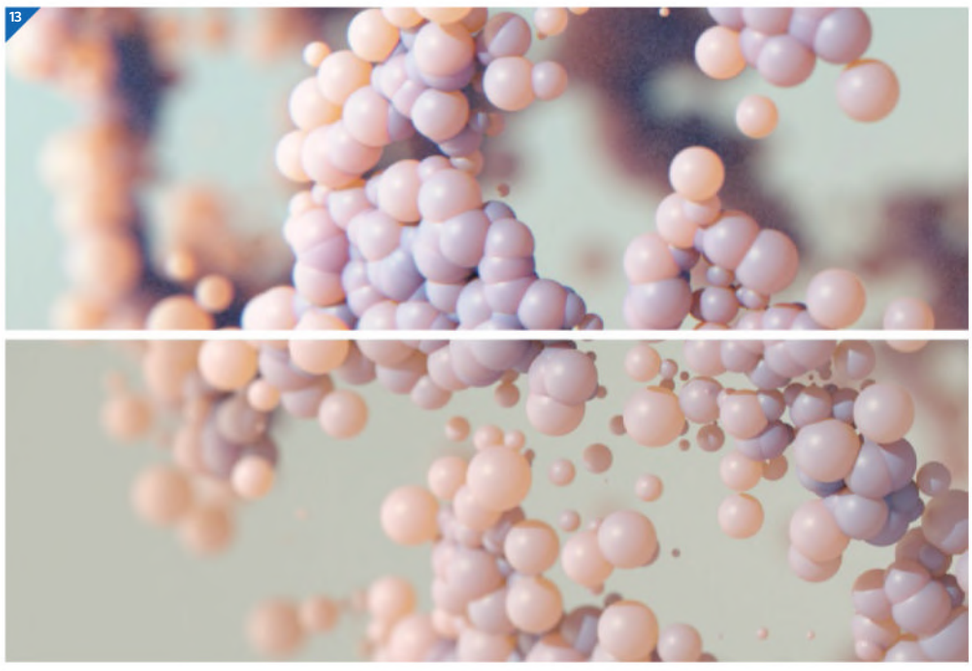
10 Creative lighting choices Adding an Octane targeted Arealight was a decision made because we wanted to have a sense of directional lighting and push more light into a specific area of the particles. To do this, select the sphere, and then add an Octane targeted Arealight. The light will always point at the sphere. You can adjust the temperature of the Octane Light to be warmer or cooler, and the power and distribution of the light to be stronger or weaker. If you deactivate the Daylight, you can visualise how much the area light is contributing.

11 Make creative adjustments Now that you have your lighting set up, go back and make some adjustments to your materials, camera and composition. Take time to fine-tune the overall look while reviewing some of your previous decisions. Maybe you'd like to make the turbulence slightly smaller? Maybe your colours are too bright? Is your sun too bright? Where does your eye go to? These are all good questions to ask yourself, when preparing for your final render.





12 Rendering considerations OctaneRender has many render kernels to choose from. We usually choose Directlighting and the GIDiffuse option. This isn't unbiased path tracing, but it provides very realistic results quickly, and because this is an abstract render, we can use it without being worried about exactness. You can increase the Specular, Glossy and Diffuse depths in the Octane Settings panel to improve the way light bounces in the final render. Since we were using a material with subsurface scattering, we increased the Diffuse depth to 8 so that light would interact with the particles in a more convincing manner. Again you can experiment with the different settings to see what works. Lower settings speed up render times, but reduces quality. Hit render and then let's head over to Adobe Photoshop.



13 Add imperfections and finish with final tweaks In Adobe Photoshop, add things like lens distortion, chromatic aberration, and vignetting to emphasise the imperfections found in real-world cameras. Most of these options can be found under the Lens Correction filter dropdown. Colour grading an image is always a subjective matter, and a very creative choice. What you choose to do with you colour will say different things about your image. In general, a less vibrant image seems more real. Also, there is no such thing as pure black in the world, and every black has a touch of colour to it. Finally, adding dust, light leaks or other textures in Photoshop helps to create a sense of space and depth. Dust can be created with a soft-edged brush with high scattering settings, or with hi-res images of scanned dust particles from the internet.

Change the lighting, change the mood

Lighting is a critical part in any render. Take some time out to understand it, and how it can help emphasise shape and form, while also making an impact on the overall mood of your image. Understanding terms like 'low-key', 'high-key', 'rim', 'key', 'fill', and other photographic lighting terms will help you to understand what lighting changes you can make in your images to make them stand out.

HOUDINI 14

Houdini, Mantra

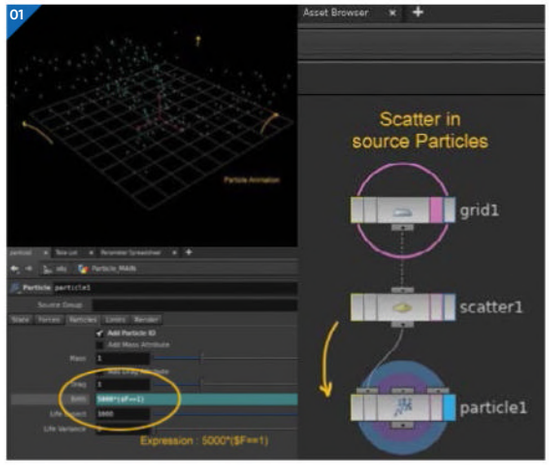
Niels Prayer
nielsprayer.com

Supamonks Studio's FX supervisor, Niels, has also previously worked at Illumination Mac Guff

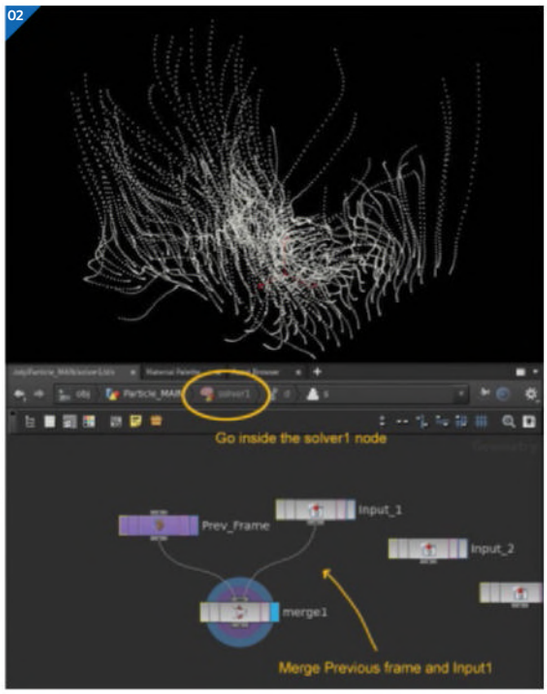
Master a particle effects system

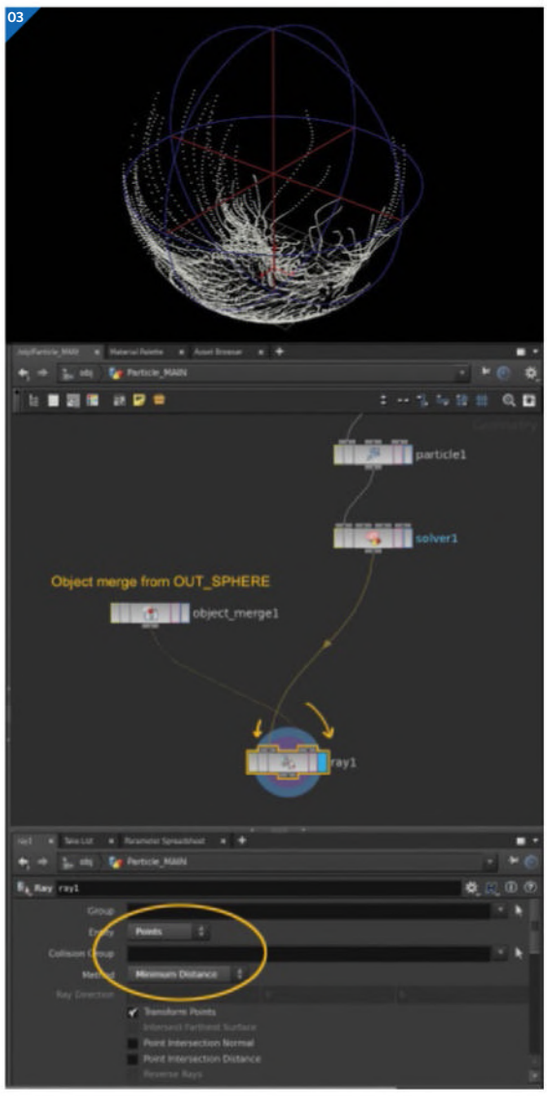
For this tutorial, we are going to create an epic motion design effect in Houdini and Mantra. This effect makes it appear as if a sphere is growing in a sci-fi style. The influence for this are the effects that can be seen on Faora-UI in *Man Of Steel*, in terms of movement and concept. From the technical aspect of this tutorial, you have to know the basics of Houdini (create nodes, working at different levels, UI and so on) but we will explain everything else needed for this effect. Our whole simulation will be based on particles and transforming in trails to create some nice curves that move really smoothly. We will learn how to create a particle system and particle trails, convert them into curves, make some manufactured objects with really sharp edges, convert edges in curves as well as how to attach geometry to points easily. We will also look at how to set up Mantra surface shaders and how to render them via the render engine. The really fun thing about this setup is the whole procedural system of Houdini: even if you have finished the work, you can easily modify the main simulation and everything else will be modified too. For motion design, especially when we have to do a lot of research during the process of fabrication, it's really important to stay procedural at each step; and that's what we have done for this tutorial. Beyond this tutorial though, you can develop your own setup based on the process used to make this effect to create a lot of cool things; the whole setup is customisable and reuseable ad infinitum.

01 Create main particles animation Create a geometry node called 'Particles_MAIN' and this will be our main simulation. Delete the file SOP and create a grid with 5 in x and y. Put down a scatter node with 250 points in Force Total Count. Create a particle node and put the scatter in the first slot. In the Particle1 node, in Forces panel, set Turbulence to 10 in x, y and z; Turb Period to 2. In Particles Panel, check Add Particle ID and put the following in Birth: '5000*($F==1)'. Now run the simulation for a smooth particle animation.



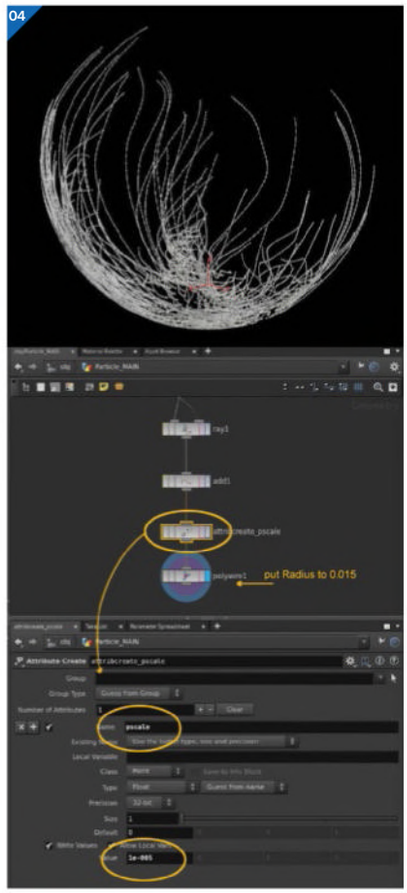
YOUR FREE DOWNLOADS
from **filesilo.co.uk/bks-798**
- Tutorial screenshots
- Maps
- Video tutorial





02 Bake particles animation with SOP Solver

Now, we have to create some nice trails for our simulation. To do that, we have to freeze the path of the particles in time. Create a solver node and plug the particle in. Dive inside the solver and create a merge node. Plug the Previous Frame slot and the Input_1 slot into the merge. Go back to SOP level. Now, if you go back to the first frame and run the sim, we have nice trails of particles growing with time with particles frozen along their own path.



03 Project with Ray SOP

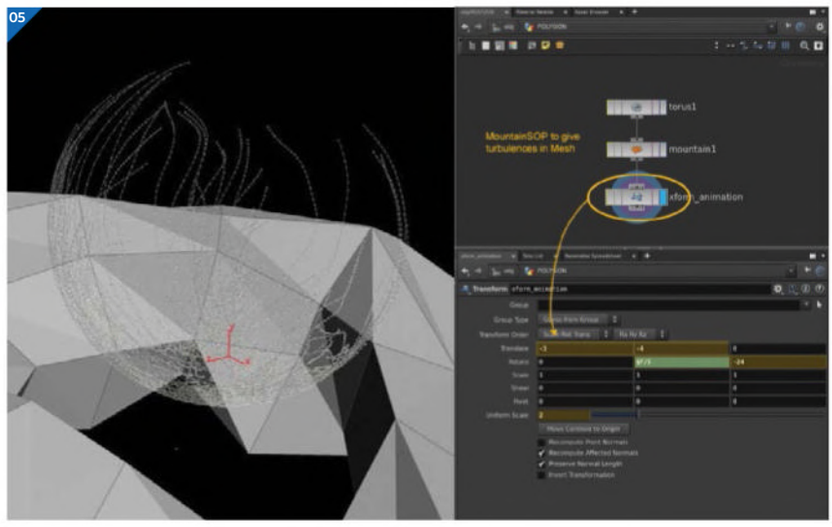
At the geometry level, create a new sphere node. Inside, put 5 for Radius. Now, with a transform, place the sphere around 3.9 in the y axis. After this, put a Null here and name it 'OUT_SPHERE'. Go back into our Particle_MAIN node and create an ObjectMerge. Then, in Object1 slot, grab your OUT_SPHERE from your previous sphere node. Create a Ray SOP and plug your ObjectMerge in the second slot and the Solver sim in the first one. In Ray, choose Points in Entity and Minimum Distance in Method. Run the simulation. You should now have your trails projected onto the sphere.

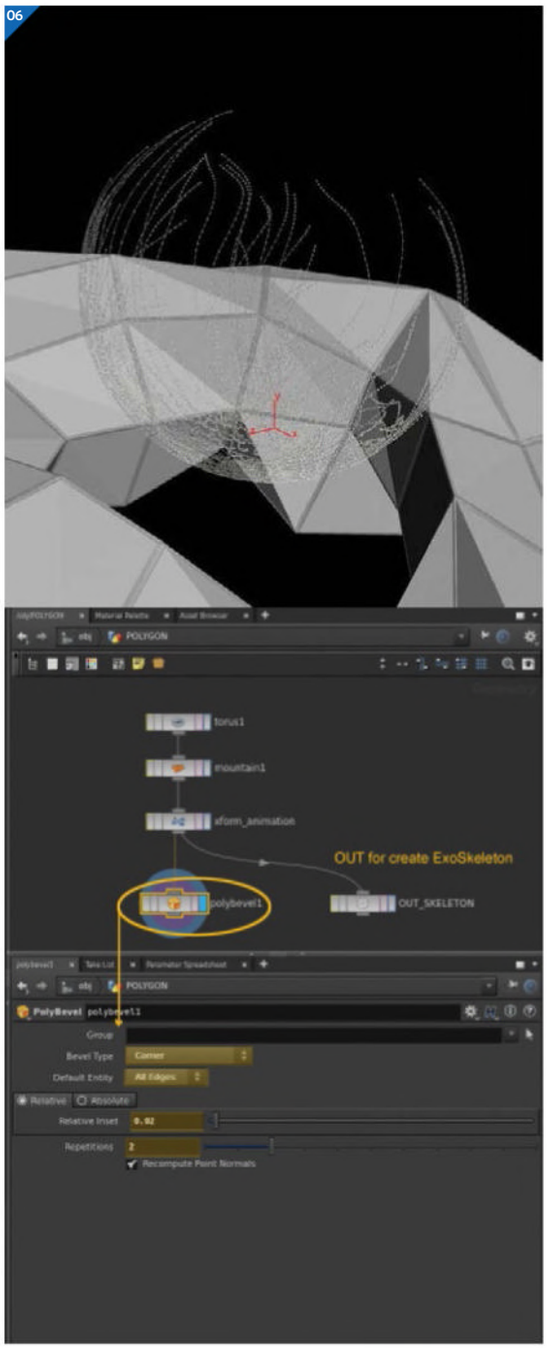
04 Convert in curves: Add SOP

Create an Add node and connect it to your Ray SOP. In the Polygons tab, check the Group panel and then set By Attribute for the Add method and 'id' for the Attribute Name. You should see your particles linked by curves. After this node, bring an AttributeCreate, set the Name to pscale and put a really low value here of around 0.00001. With that, only the curves will be visible at render time, not the points (the scale will be really small). Finally, put a Poly Wire after that, and set the Radius to 0.015. Now your curves are converted to polys and ready for rendering!

05 Create a background object

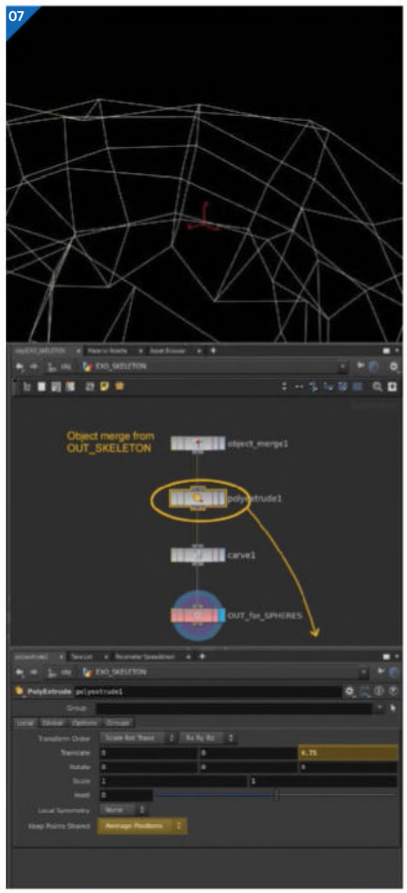
Now let's leave the sphere curves for a moment and work on something else. At object level, create a new Torus object. Set the Primitive Type to Polygons and Radius to 5 for x and 2 for y. This is where we will set eight rows by 11 columns. After that, add a Mountain node, and set the Roughness to 0.95 and the Height to 2. Now, with a Transform node, place your shape right under your previous sphere curve object, around -3 for x and -4 for y. Set the Uniform Scale to 2. Now you can put the expression $F/5 in Rotate Y for a nice slow movement to the shape.





06 Add details to polygonal object

Now we have to add details to this shape. To make nice flat faces, we need a PolyBevel right under our Transform. Set the Bevel Type to Corner and the Entity to All Edges. Stay in Relative mode and set the Relative Inset to 0.02 and the Repetitions to 2. Now, we have really sharp faces and nice bevels around each faces which give a really manufactured aspect to the object. Then, add a Null after the transform (next to the PolyBevel) and call it 'OUT_for_SKELETON'. We will use the unbevelled shape for the exoskeleton shape later. Now display the PolyBevel node.



07 Exoskeleton around the polygonal shape

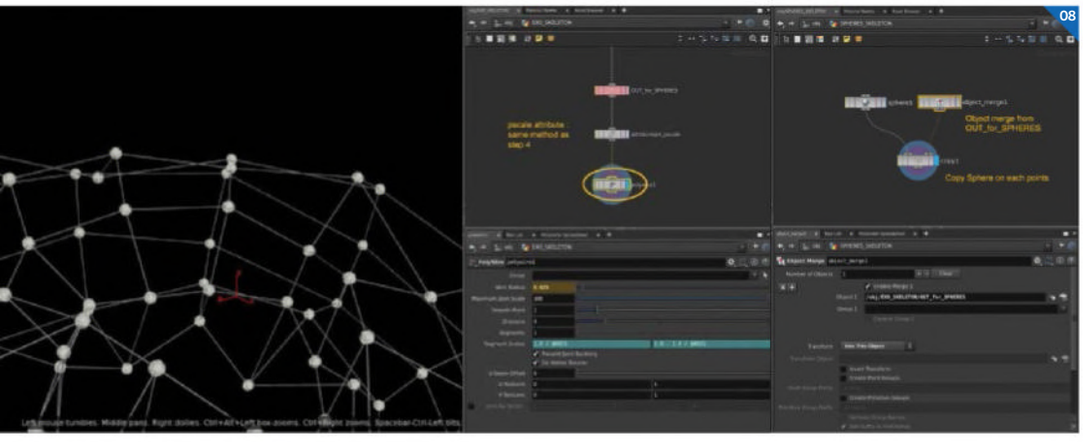
At the object level, create a new Geometry object. Call it 'EXO_SKELETON'. Inside, delete the file node and create an ObjectMerge. Find your previous OUT_SKELETON Null in Object1. Add a PolyExtrude to it and set the TranslateZ to 0.75. Set Keep Points Shared to Average Positions. After that, add a Carve SOP to keep only a contour of the shape. Put a Null right after the Carve and name it 'OUT_for_SPHERES'. We will use that later to put spheres on the points of the shapes.

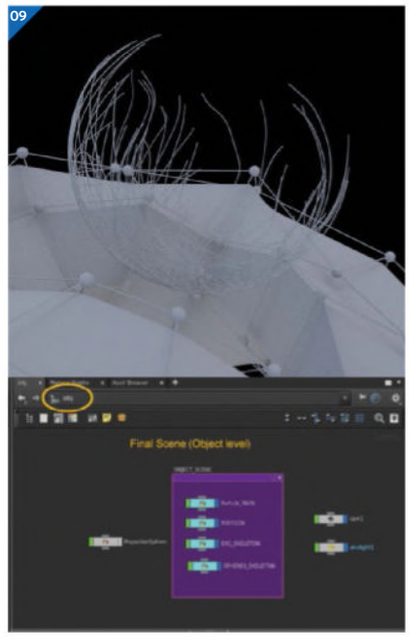
08 Detail refining on the skeleton

First, set an AttributeCreate SOP and in Name put 'pscale'. Then, put the Value at something like 0.0001 (in the same process as before), so that points are invisible at render time. Now, put down a Poly Wire node and set the Wire Radius to 0.025. Now, at the object level, create a new Geometry SOP with a ObjectMerge into it set to OUT_for_SPHERES. Next to it, add a Sphere with 0.25 for Radius. Put down a Copy SOP and plug the Sphere in slot 1 and the ObjectMerge in slot 2. Now our exoskeleton of tiny spheres is finished!

09 Set up lighting and camera

Now we have all our objects created. If you run the sim, you should see that the curves have created a sphere with your shape slightly turning under it on the y axis. We now need a camera for the scene. At the object level, create a Camera object and place it like the image below. After that, create an environment light. In the Light Panel set the Intensity to 2 and put an HDRI in the Environment Map. Now you should have your scene set and lit, prepared for shading and rendering.



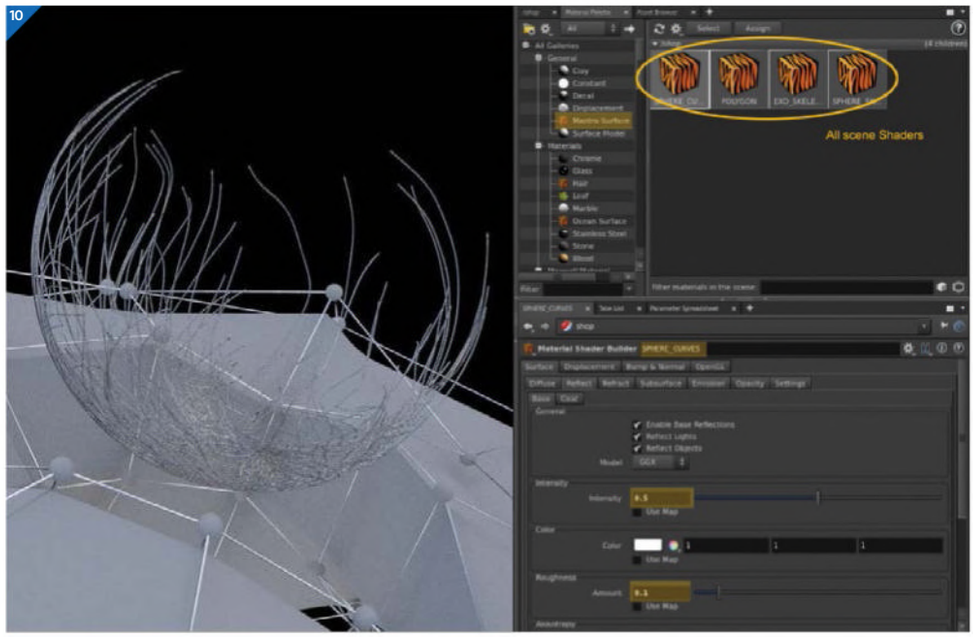


10 Shading with Mantra and rendering First, go to the toolbar and create a PBR render through Render>Create Render Node>Mantra-PBR. In the Material Palette, create four Mantra surface shaders. Now assign one shader to the Particle_ Main object, one to the POLYGON shape, one to the exoskeleton and finally, one to the SPHERES_SKELETON. Rename them properly. For the sphere curves, we need a chrome metal shader, so set the Diffuse intensity to 0 and the Base Reflection Intensity to 0.5. You can also apply the EXO_SKELETON settings onto the Mantra surface.

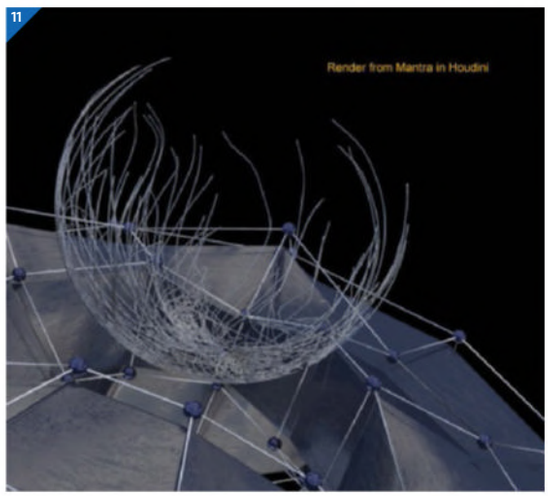
11 Further shading values Now we need to set up the other shaders. For the Polygon object, you can set the Diffuse intensity to 1 and the Base Color to a dark purple. In the Base Reflection, leave the Intensity at 0.1 and set the Roughness to 1. Check Use Map and put your Roughness map in the slot to make a nice-looking specular. Now, for the SPHERE_SKELETON one, set the Diffuse Intensity to 0.1 and give it a dark blue colour. Then in the Base Reflection, set the Intensity to 0.5 with a grey-blue colour.

12 Set the render and export In the Network view, go to the out panel and click on Mantra1 node. Set the frame range to 1-150. In the Images Panel set your output path to your output directory and set the Camera to cam1 (yourFolder/name.$F4.exr). Just below, in the Extra Image planes, check Shading Depth Pz so that we have a Z-Depth pass in our EXR. Now, in the Rendering panel, set Pixel Samples to 7 by 7. Finally, go to the top of the menu and hit Render. You should have an EXR with Color, Alpha and Z-Depth for comp.

10



11



Compositing tips

You could create a really nice image with your three passes. You can increase contrast to give more impact to the image for instance. Then, you can add a little glow on the highlights to give a bloom effect to the scene, and with the Z-Depth map, you could add a nice depth of field with nice bokehs using the frischluft plugin in After Effects for example. In the background, it's better to have something really blurred to give it a nice depth, but without a lot of information, so that we stay focused on the foreground effect.

12

3DS MAX

3ds Max, FumeFX

Vikrant Dalal
project01studio.blogspot.in

Vikrant has over eight years of experience in the VFX industry. His company Project01 Design Studio works with international clients and produces 3D tutorials

Create a tornado effect

Tornadoes are very interesting-looking weather phenomena, they can vary in shape, size and are very much dictated by movement. In this tutorial we are going to take a look at how you could make one, like those seen in films such as *Twister*, *The Day After Tomorrow* and *Into The Storm*.

There are a number of great plug-ins for 3ds Max that can create a decent tornado effect, such as FumeFX, Afterburn and Phoenix FD. Many larger visual effects and animation studios use FumeFX software in their processes, as this software gives good output, is trustworthy and, most importantly, is user friendly.

To create this effect, we are going to use 3ds Max and FumeFX plug-in. There is quite a lot to learn in order to get to grips with each and every parameter of this software and it would take a lot of time to go through. As far as the tornado effect is concerned, we will learn as much as required for this particular tutorial only.

There are different techniques you can use to make this effect in FumeFX. Simple Source, Object Source or Particle Source, but we are going to use Object Source.

Before you start working on this kind of effect, you should have a good knowledge of real-world scale. You must know the height and diameter of the tornado, the rotation speed, the amount of wavy movement and the speed it travels from one place to another – a combination of all these factors will decide the scale of tornado. There are many different types of tornadoes that form in real life, for example landspout, multiple-vortex tornado, waterspouts, supercells, dust devils, gustnadoes and many more. So you should check which type you are referencing first before you start work on the shot.

This is very interesting subject because you can't define one certain process to create this effect. It very much depends on your own understanding of using tools and techniques. As you use your creativity and tools, you will find different types of effects every time. So let's get started!

YOUR FREE DOWNLOADS
from **filesilo.co.uk/bks-798**
- Tutorial screenshots
- Video tutorial

01 Create and animate the spline

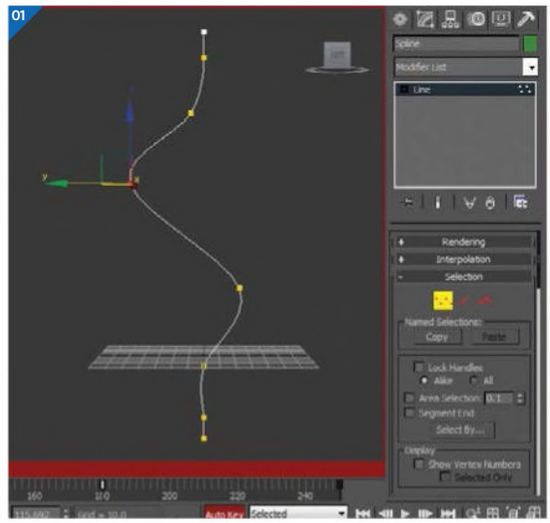
To achieve our tornado effect, we need to create the base model of a tornado, because we are going to animate the object in the same way as a real tornado. We will use two cones and one spline; one cone will come from the top and one from the bottom to meet in centre. To animate these cones we will use a spline. So, to create the spline go to Create> Shapes> Line. Draw a straight line, go to the Modify panel and add three to five vertices with the help of Divide. Turn on the Auto Key and animate every single vertex. This motion should look like real tornado behaviour. There are two ways to animate this spline, the first is Manual Animation, which we have used, and the second is to select all vertices except the first and last, then assign the Noise modifier on top of them. Turn on the Animation option and adjust it as you require.

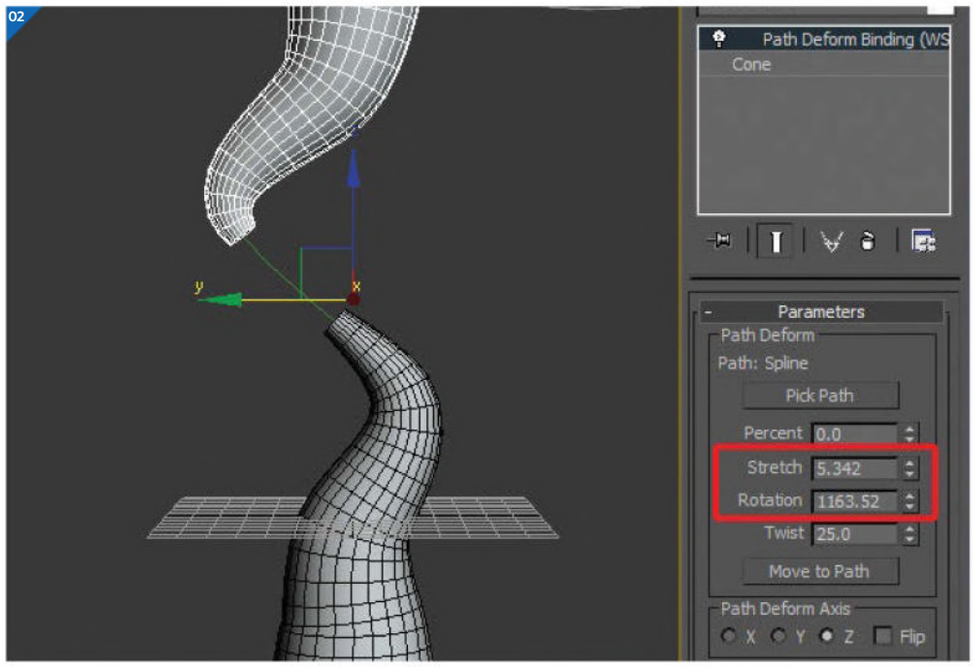
02 Create cones and animate them

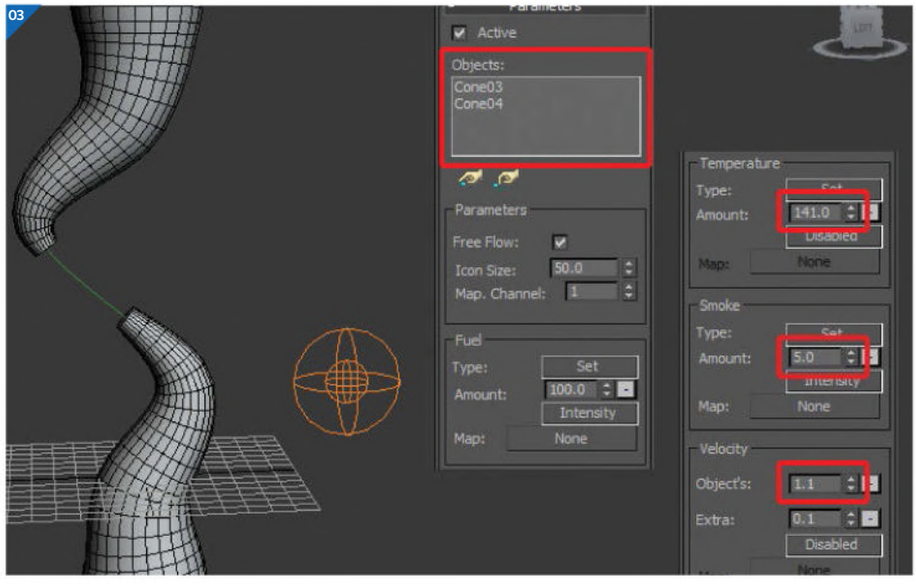
In this effect we want to show two tornado parts coming together to form one tornado. Similar to actual tornado footage, one tornado comes from the clouds and the other generates from the ground to merge in the centre. To create this effect we will use two cones. Create cones from standard primitives, then add a Path Deform Binding modifier to both cones. Then click Pick Path and select the animated spline, which we have already created. Now animate the stretch parameters as per the requirements. We need to animate the Rotation Parameters too, because real tornados continuously rotate in one direction. So after you've finished the animation and editing of the parameters, we can see that our base model (cones) is ready to generate the smoke. You can use my 3ds Max File with the base model, that is free with this issue.

03 Create FumeFX object source

Now it's time to create our important element, which is the 'smoke'. To create smoke, we will need two things: one is FumeFX Object Source and the other is FumeFX Container. With the help of FumeFX Object Source, we will generate and control the fire and smoke that emanates from the objects. To create FFX Object Source, go to Create Panel > Helpers > Click on dropdown menu > FumeFX. Click on the Object Src button and generate in viewport. Now rename this 'FFX Object Src' 'Tornado_Emitter', now select it and go to modify panel and add our both cones in to Object tab. Don't change anything in Fuel, just change the Temperature with the amount of 140, Smoke 5.0 and Object's Velocity 1.1. This is as simple as that.

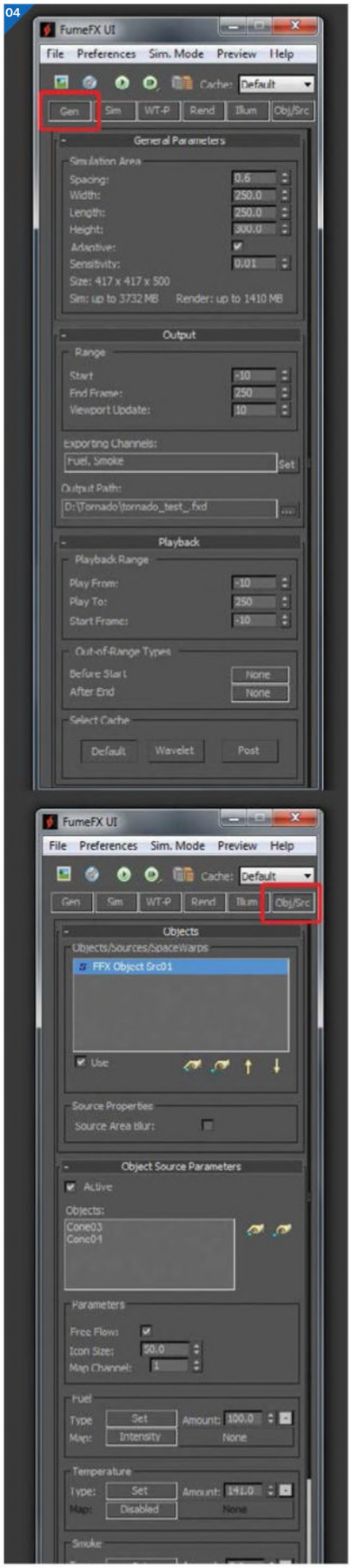


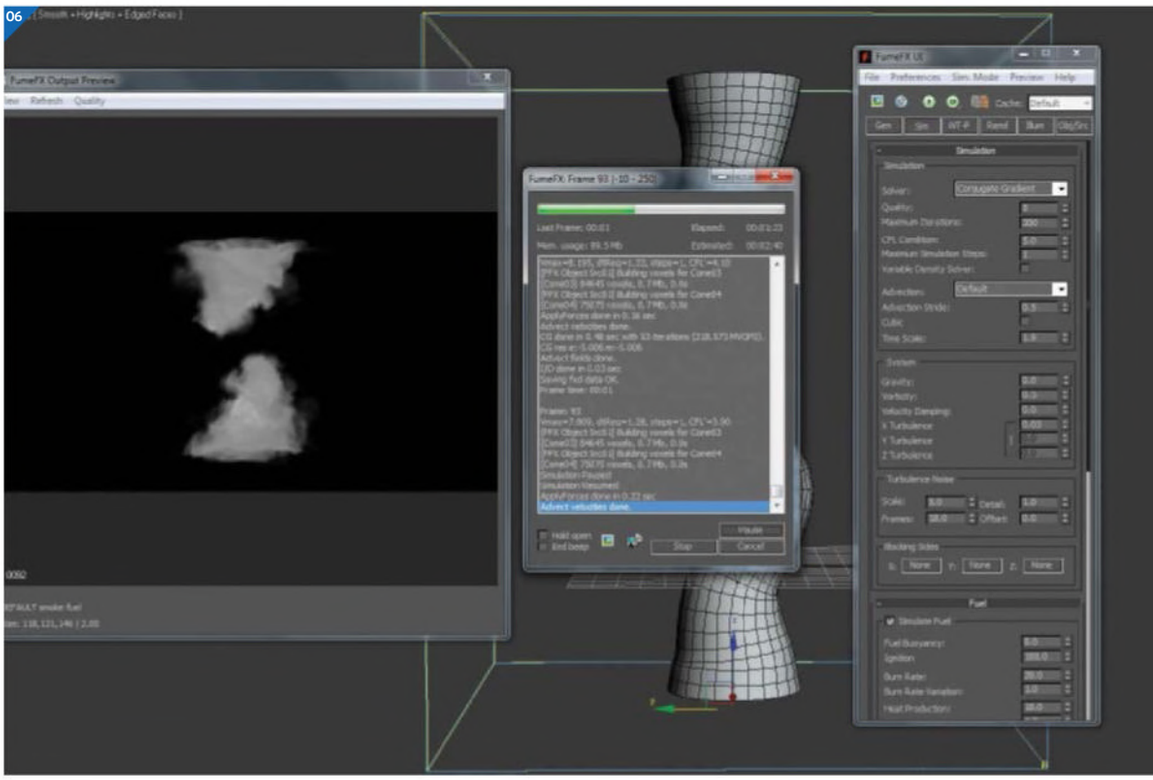




Create tornado using Particles

In this tutorial we have used FumeFX Object Source. However, if you're capable using Particle Flow System, then you can also create effective output. You can use various space warps to control particles and give it a good shape, so it's easy to control particles for things like speed, rotation and turbulence. Then you simply have to add these particles into FumeFX Particle Source. If you are using FumeFX Simple Source or FumeFX Object Source then you can use Space Warps. Use Wind Space Warps for turbulence and Vortex Space Warps for tornado rotation, but remember, you should keep the parameters high for the Wind or Vortex while working in FumeFX compared to working with Particle Flow System.





04 Create FumeFX Container Let's create our FumeFX container. To create this container, go to Create Panel> Geometry> then click on the drop-down menu and select FumeFX. Next select the FumeFX and generate the container in viewport, then modify its parameter. Let's start with General Parameters; make the Spacing 1.0 and the Size 250x250x300, then set the output and playback range from 0 to 250. Then navigate to Output Path, make a new folder on your storage drive and save your cache file as 'tornado_test' in the new folder. After you have finished the editing of general parameters, go to 'Obj/Src' and add our 'FFX Object Source' in to object tab. Once this is done, we can move ahead to the simulation section.

05 FumeFX simulation parameters In the simulation section, begin by making the following settings. Keep everything as it is except Time Scale – set this to 1.9. Now move ahead to System. Set Gravity to 0.0, Vorticity to 0.3 and Turbulence to 0.03. For Turbulence Noise set Scale 5.0, Frame to 10.0 and Details to 1.0. All of X, Y and Z should be set to none in blocking sides. Now go to Fuel Parameters and keep everything as it is. Last of all, set the Smoke Dissipation Min Dens to 7.0 and Dissipation Strength to 10.5.

06 Start the simulation We have already made all the necessary changes in the FumeFX Object Source, so there is only one thing left to do: the simulation itself. So just hit the Simulation button. It will take around six to seven hours to simulate 250 frames. It all depends on the details of smoke and scale of the scene.

Use real footage

In this tutorial, we have created our tornado on a plain black background. But you don't have to stop there! To make the tornado look even more authentic, shoot any live scene using a handy cam and try to merge your tornado into that footage. From here you can create a complete visual-effects shot, from shooting, camera tracking, FX and compositing. Also remember to add extra elements to your scene, for example dust, debris and other objects.

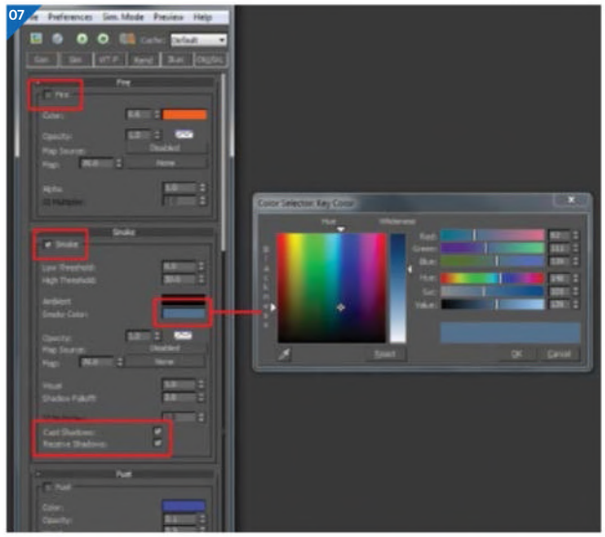
> For the final rendering of the smoke effect, we will use the Default Scanline Renderer, as we are not going to apply Global Illumination, Light Scattering or Light

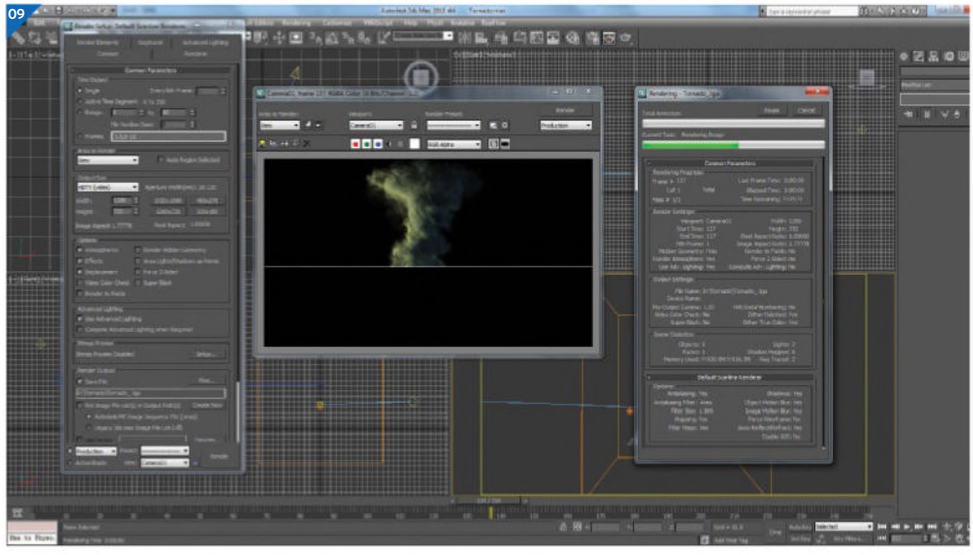
07 Tweak FumeFX rendering parameters After the simulation is finished, it's time to assign some colour to the smoke. We will turn off the Fire tab, because we only want the smoke element, so let's start with ambient colour. Set the ambient R:0 G:0 B:0, with the smoke colour set to solid colour, make it R:88 G:111 B:139. The opacity should be set to 1.0 and both the cast and receive shadows boxes ought to be checked.

08 Modify FumeFX illumination parameters Before moving forward to the next steps, we need to place the lights. We are going to use two lights. One is Omni and the other is Target Spot. For Omni, set the position as X:-530, Y:-390, Z:158. Turn on shadows, then go to Shadow Parameters tab and turn on the atmosphere shadows with Multiplier set to 0.4 and Color set to R:220, G:255, B:255. Keep everything else as default. For the Target Spot light, set the light position to X:180, Y:550, Z:50 and Target position to X:0, Y:-10, Z:45. Like before, turn on the Shadows and go to the Shadow Parameters tab to turn on the Atmosphere Shadows, with a Multiplier set to 2.5 and Color set to R:255, G:191 and B:91. Again, keep everything else at the default setting. Now go ahead and add these lights to the FumeFX Illumination tab.

09 Rendering For the final rendering of smoke, we will use the Default Scanline Renderer, as we are not going to apply Global Illumination, Light Scattering or Light. To render the smoke element, press F10 and the Render Setup window will appear. Set the Frame Range and Output Resolution as per your requirements. Now save the Render Output Path and hit the Render button. Wait a little while and after some time, we'll get our smoke element.

10 Post process with After Effects We can now composite the smoke element. Load this layer in to the Timeline and assign some effects to it. Let's start with Color Balance. Navigate to Effects> Color Correction> Color Balance. Now set the Shadows to R:-19, G:5, B:46. Set MidTones to R:32, G:16, B:1 and Highlights to R:43, G:-18, B:-9. Once again go back to Color Correction and select Brightness and Contrast, set the Brightness to 13.0 and Contrast to 0.0. After this go to Effects> Blur and Sharpen. Select Sharpen, set it 50. You can use your own effects or parameters to give it a more realistic look.





MAYA

Disintegrate a model with nCloth and nDynamics

Maya
Amaru Zeas
www.amaruzeas3d.com

Originally from Cuenca, Ecuador, Amaru is a passionate 3D Artist based in Seattle

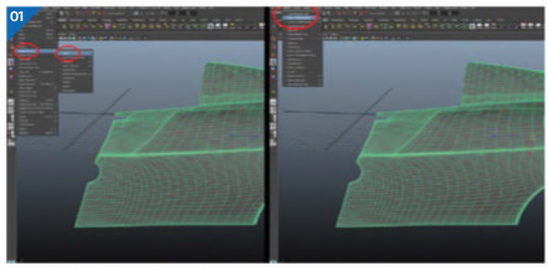
There are so many ways we can make our renders more interesting and creative. You would be surprised how quick and easy it is to create amazing effects using Maya's nCloth and nDynamic Force Fields such as Newton or Volume Axis. nCloth is a dynamic cloth solution that uses particles to simulate dynamic polygon surfaces. You will find nCloth useful for simulating fabric clothing, shattering surfaces and deformable objects. Now, to make the particles move we will need to add a force and Maya has a powerful nDynamic system for this purpose. You can use nDynamic Force Fields and Point Force Fields to generate fields to push or pull Nucleus objects. There are several Force Fields that we can use like Drag, Newton, Vortex, Volume Axis and more. In this tutorial we will be using Volume Axis just because it works perfectly with the effect that we want to create. Volume Axis has incredible attributes that we can play with. This tutorial will work using any polygonal object and the poly count doesn't matter, however keep in mind that using high-polygon models will increase the time taken for you to complete the tutorial. There are also several ways to create the same effect and you can find them online if you wish.

There will be six steps that will help us in this tutorial. We will start by setting the scene and then we will go over nCloth. After having a better understanding of how nCloth works, we will go over nDynamics and Field Forces. Finally we will set the scene to create the final simulation.

01 Set up the scene In Maya create a new scene, File>New Scene, just to make sure that we start with a clean setup. Open or import a polygonal object that you want to work with. It doesn't matter how complex the object is or its topology. We will be working with a piece of a car that we have modelled for this tutorial. Select the object, freeze transformations and delete history by going to Edit >Delete by type>History and to Modify>Freeze Transformations.

02 Detach polygon faces Now separate the faces of your object so that we can make them move individually. We will use Detach Component (located in the Polygons menu under Edit Mesh). Select your object (note that any users with Maya 2014 or older might need to go to Vertex mode before using Detach). Now go to Edit Mesh>Detach and click. You will see that the object's vertex is highlighted. Also on your object you can go to face mode, select a face and it should now move individually from your object.

03 Make the object nCloth It's time to covert the object into an nCloth and change the Menu set to nDynamics. Select your object then go to nMesh>Create nCloth. This will create an nClothShape and a Nucleus. You can



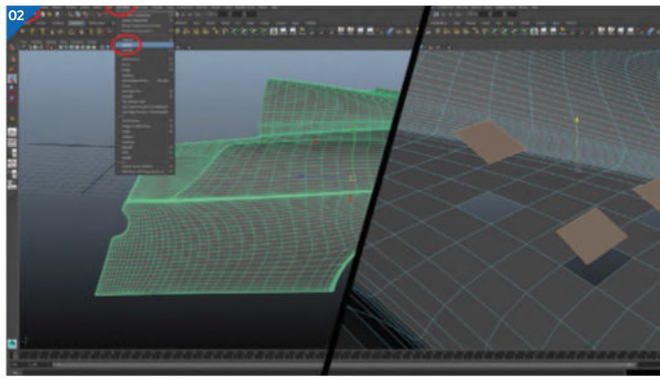
YOUR FREE DOWNLOADS
from filesilo.co.uk/bks-798
• Tutorial screenshots

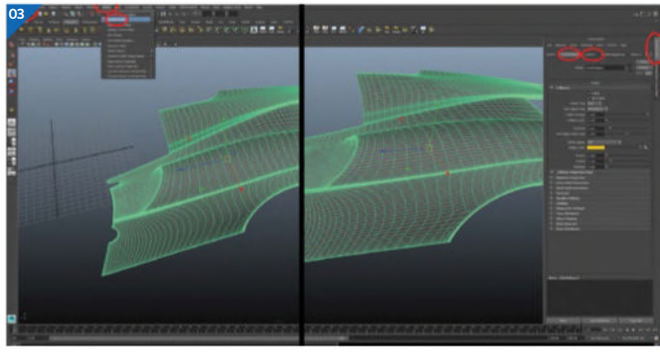
see the new attributes under Attribute Editor by selecting the object. The nClothShape and the Nucleus have different attributes and we will be using them back and forth until we get the desired effect on the nMesh.

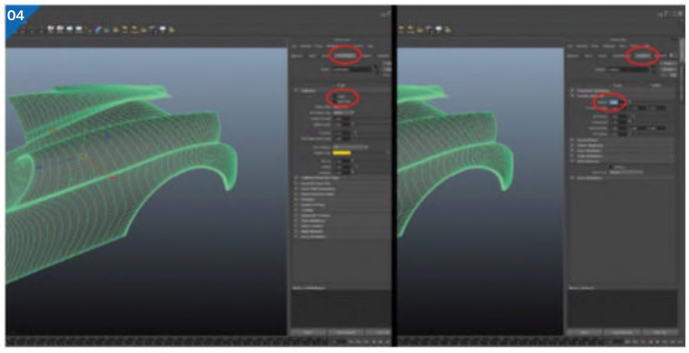
04 nClothShape and Nucleus attributes Open the Attribute Editor by selecting the object. We are going to set up three main attributes on the nClothShape and Nucleus so that the simulation will be faster and more efficient. First, click on the nClothShape tab then open the Collisions attribute. We want to uncheck the Collide and Self Collide option box. If you want you can leave Collision on but the simulation will be very expensive. Lastly, go to the Nucleus tab and under the 'Gravity and Wind' attribute set Gravity to 0 since we want only the Field Force to have any effect on the object.

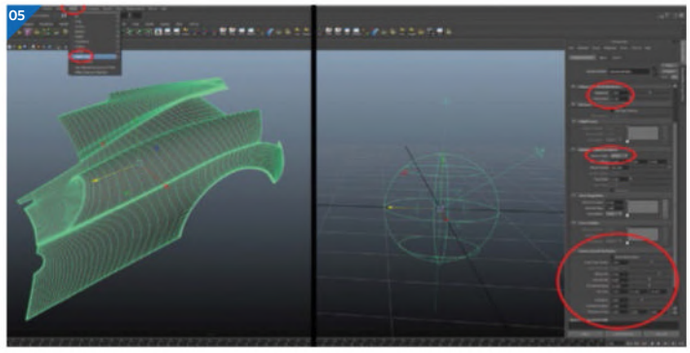
05 Add Force Field with nDynamics We will be using Volume Axis Field so select the object and go to Fields>Volume Axis. We might be changing quite a few attributes and we will do this by selecting the field that you just created, going over to the Volume Control Attributes and changing the Volume Shape to Sphere. Under Volume Axis Field Attributes set Magnitude to 1 and Attenuation to 0.1 (this will be for falloff). Directional Speed should be set to 10 and Direction on y to 20 with z at -30. Under Volume Speed Attributes set Along Axis to 5. Let's play with the Turbulence by setting it to 3, Turbulence Speed to 1, and Turbulence Freq x y and z set to 4.

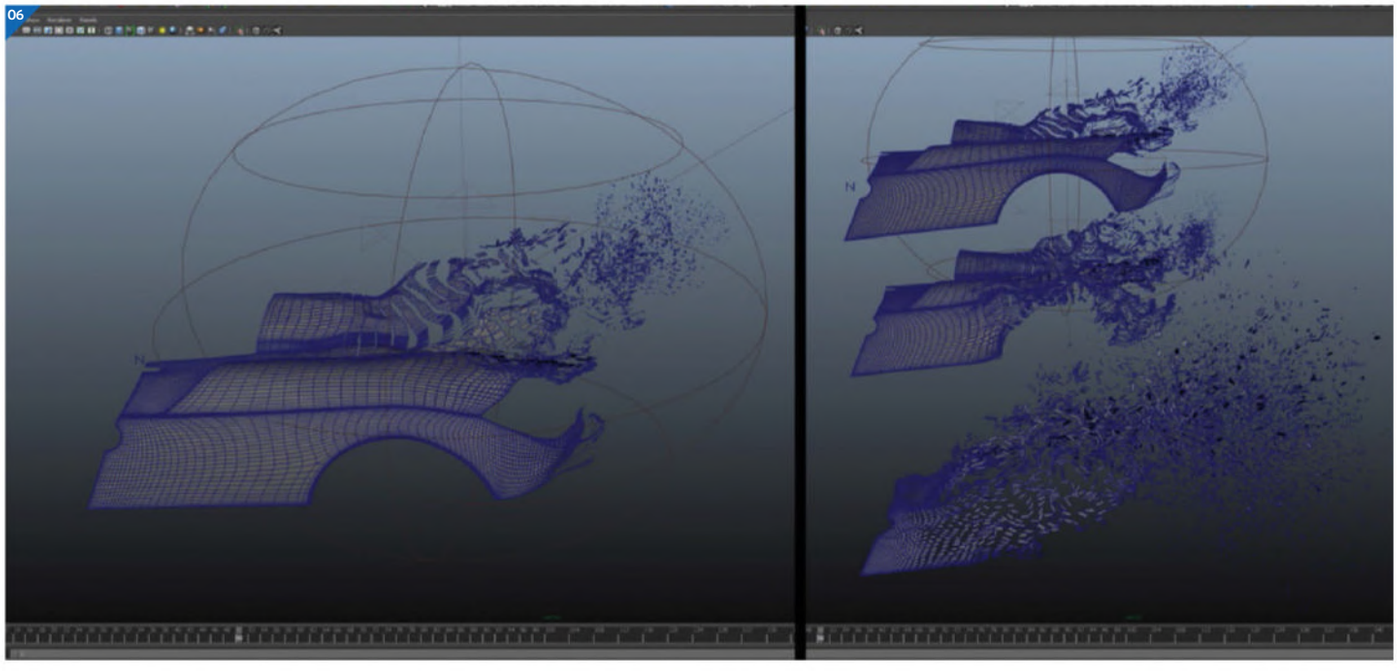
06 Get a simulation You can see that the field has two arrows pointing at different directions and the particles will move to those directions. Hit play and you will see the simulation. Finally we can place the field at the back of the object. Create an animated key on frame one then go to frame 100, move the field and create another key so that the field has an animated motion. You can pause the animation and make a copy of the object to have them as a reference. Remember that our settings works well on this project but you can of course change them accordingly for your own project to achieve different results.











MAYA

An introduction to Bifröst in Maya

Maya
Dave Scotland
www.cgswot.com

Currently lecturing at AIE Australia and with years of CG production experience, Dave has worked for some of the biggest studios

How do I animate pouring liquids in Maya? If you had asked me that question a couple of years ago, my answer would have been different. I would have stepped you through Maya's built-in fluid system and then shown you all of the tricks to get it looking acceptable. However, now that we live in a world where Maya has reimagined its fluid simulation system, there is really only one method I would recommend, and it's called Bifröst.

Originally developed by Exotic Matter and using Naiad technology, Bifröst is a procedural engine, capable of generating liquid simulations. Based on a FLIP (Fluid Implicit Particle) solver, Bifröst can generate liquid from emitters, which then react to gravity and also be directed using colliders and accelerators.

Although fully integrated into Maya, the simulation calculations are actually handled externally by the Bifröst Computational Server. Using Viewport 2.0, this enables you to author, refine, playback and scrub within Maya, all while the simulation is actually being processed in the background. Such interaction is crucial to bringing a greater level of creativity and control, while still providing a very high degree of efficiency.

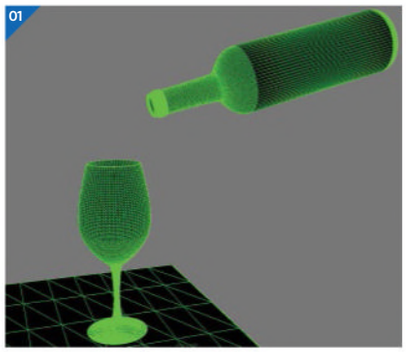
Over the following steps I'll introduce you to the Bifröst system and show how to get a basic simulation up and running. I will step you through the process using a simple scene involving pouring wine from a bottle into a glass. It's important to note that even though we are using a simple scenario to go through the procedure, the same steps can be applied to much larger simulations, such as on pools, waterfalls, rivers and oceans, to name just a few.

Our aim in this tutorial is to replicate the effect of filling a wine glass with liquid from a wine bottle and to have the liquid react naturally to the various objects within the scene. For the purposes of generating a final render, we also want the materials to replicate an actual liquid. To achieve this goal, we'll create a simple geometry emitter to represent the wine coming out of the bottle. We'll set both the bottle and the glass as collision objects, so the liquid will react to these objects correctly. We will also set up a Kill-Plane to increase the efficiency of the solve.

Once we are happy with the simulation, we'll bake out the information to a cache sequence and create a liquid mesh object, using the baked data. This mesh can then be textured, using the Bifröst material for the final render.

YOUR FREE DOWNLOADS
from **filesilo.co.uk/bks-798**
- Scene files
- Video tutorial
- Tutorial screenshots

01 Prepare the scene It is important to set the scene up correctly before starting the simulation process. This means animating any objects that need animation and correctly positioning everything in the scene. It's also important to get the scale right. Bifröst works exclusively in metres, regardless of the scale units currently set in Maya. For example, if you have a cube 100 units high and your Maya units are set to centimetres, Bifröst will treat the object as 100 metres high. This is a critical element to remember, as any simulation will not look right if the scale is not correctly planned out.

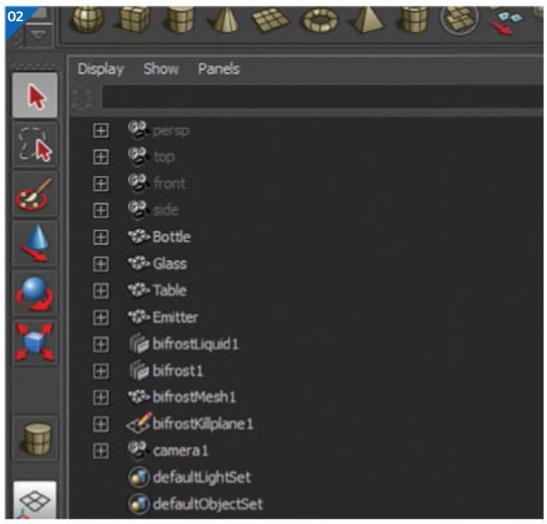


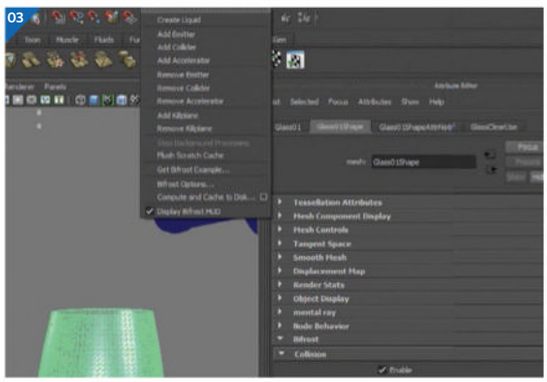
02 Create the Bifröst simulation objects Let's start by creating the emitter object. We will use a simple polygon cylinder carefully placed in the neck of the bottle. Scale it so there is clearance around the emitter and is not penetrating into the bottle's geometry. With the emitter selected, go to the Bifröst menu and select Create Liquid. This will produce all of the required Bifröst objects, which include Bifröst1, BifröstLiquid1 and BifröstMesh1. You'll also notice the additional controls added to the emitter's shape node, under the new section entitled Bifröst. In fact, any object used as an emitter, collider or accelerator, is given extra Bifröst controls.

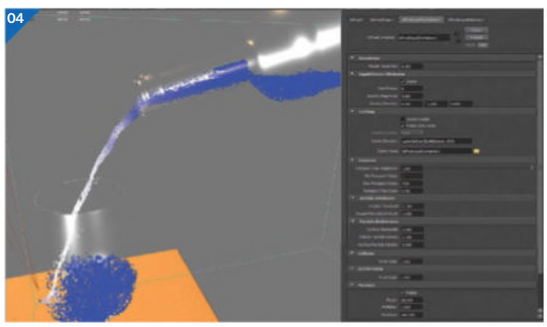
03 Make the collision objects Using the Outliner, select Bifröst1 and click the timeline's play button. You should see a growing yellow bar appear on the timeline, with a green bar growing slightly behind it. This indicates that the simulation is processing. You should also see the particles are dropping straight through the bottle. We need to turn the bottle into a collision object. Select the Bifröst1 object, then Shift-select the bottle and choose Add Collider in the Bifröst menu. Now when you press play the bottle will deflect the particles. Do the same for the glass.

04 Refine the simulation The most important setting is the Master Voxel Size, located at the top of Bifröst1's BifröstLiquidContainer1 panel. This parameter sets the size of the voxels in metres scale. For our wine glass simulation we need to be working in centimetres, so set it to 0.005. This means the voxels are 5cm in size and will create a more-realistic simulation. As well as this, it will increase the simulation processing time, but it's the price we pay for accuracy. We also want to decrease the Transport Time Scale to 0.75. This provides a more-stylistic look by slowing down the simulation time.

05 Bake out the cache Now that we're happy with our result, let's bake out the cache data to save having to process the simulation every time. This will also make the process of creating a mesh much more responsive. Simply go up to the Bifröst menu and select Bifröst Compute And Cache Options. Set the output location and cache name, then click the Create button. This will create a file sequence, which contains the simulation data. Once processed, select Bifröst1 and go to the Caching section. Tick the Enable Disk Cache option and ensure the cache path is correct.



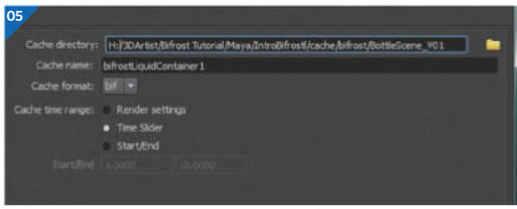


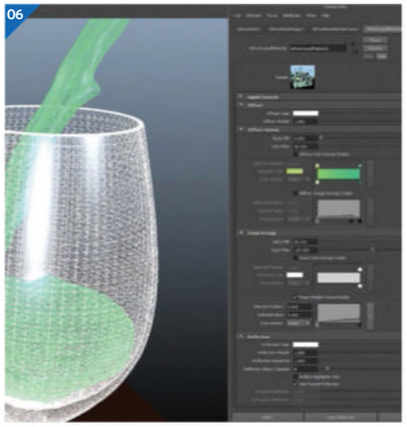


06 Make a simple mesh and material Select the Bifröst1 object and tick the Enable setting in the Bifröst Meshing options. You should now have a mesh representation of the simulation. Tweak the settings for the mesh until you're happy with the geometry, then hide the Bifröst1 object. Notice that the mesh is still visible. You are actually seeing the BifröstMesh1 object, which already has a material applied. Head over to the BifröstLiquidMaterial1 tab and adjust the settings to suit your desired effect. Try disabling the Foam Remap controls, as the simulation's timing is quite slow and would not have a great deal of foam.

Five tips for better Bifröst simulations

- **Scale it** Making sure that you use the correct scale in your scene is imperative to getting a good simulation.
- **RAM it** In the Bifröst Options, set the Maximum RAM Usage to an amount that is appropriate to your system to ensure that it works well.
- **Avoid it** To avoid crashing and other undesirable results, try your best not to animate any of the BifröstLiquidContainer attributes.
- **Fill it** Fill a pool or emit a single drop by turning off Continuous Emission in the emitter's Bifröst settings.
- **Kill it** Adding a Kill-Plane to the scene will eliminate unwanted particles from the simulation, saving valuable simulation computations.



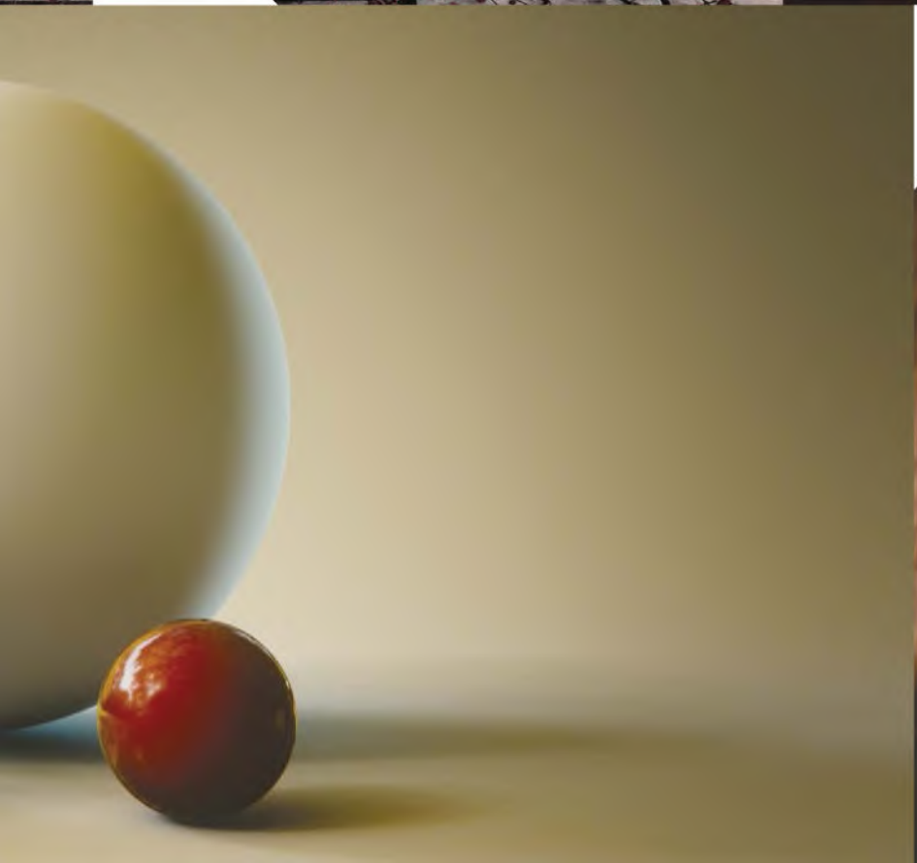


Lighting & Rendering


176


172


166




154

Set up a lighting rig in Blender

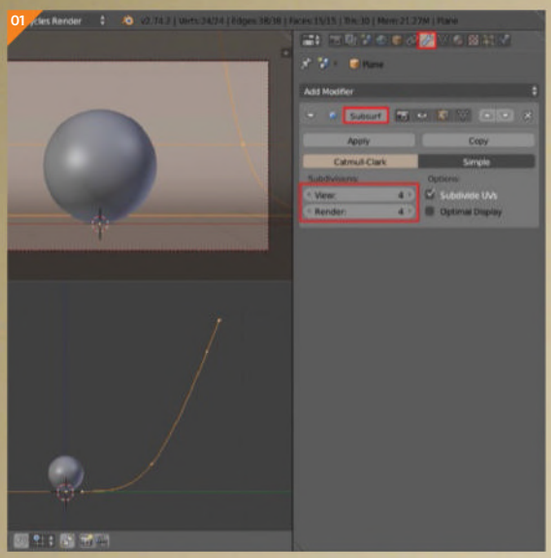
Lighting, in its broadest sense, can make or break a scene – it is often one of the most overlooked aspects of 3D and will dictate how believable and captivating your scene is. This is one of the reasons why some artists will dedicate their 3D career to understanding and deconstructing it. And with the advent of the Cycles rendering engine for Blender, lighting has never been so fun and intuitive.

With that said, there are a few basic principles and rules to follow to create convincing imagery, which will also give you more room to break these and experiment. The question of what type of lighting is best comes up often. As much as we would like to give a definitive answer to this, it all depends on what scene you currently have and what emotion you are trying to portray. It also goes hand in hand with the other aspects of your scene and should, in most cases, exaggerate and give more meaning to them.

A very rudimentary example of this is when you have gone all the way in creating a very detailed model that has very convincing and realistic textures applied to it; however, during the lighting stage, you decided not to give much emphasis to it and just dropped in a few lamps here and there. After a while, you realise your scene isn't looking as good as you had hoped it would be, despite your efforts of adding all the necessary modelling and texturing details. Disappointed, you decide to abandon the project and swear to never finish it.

In this brief guide, we'll touch on the fundamentals of lighting and how you can incorporate them into your scene and avoid caveats that may demotivate you.

01 Basic scene setup For this guide, we will be utilising a very basic scene setup: a sphere and a backdrop. This is most useful when gauging your lighting scheme and will serve as a great reference later on when attempting larger and more complex scenes. Add a UV Sphere with the default settings, add a Subsurf Modifier level 2, and set the shading to Smooth. For the backdrop, add a Plane with a Subsurf Modifier level 3, then extrude the back part, as seen in the image below. Next, place your camera in front of the sphere so you can clearly visualise the whole scene.



Blender, Cycles

Reynante M Martinez
reynantemartinez.com

Reynante specialises in visual storytelling and lighting, academic teaching, art direction and arch-vis

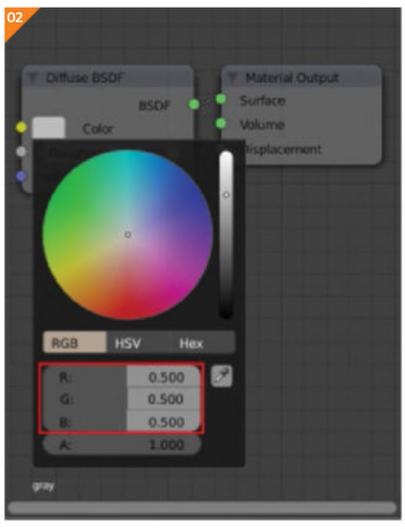
YOUR FREE DOWNLOADS
from **filesilo.co.uk/bks-798**
- Tutorial screenshots
- Video tutorial


Some artists will dedicate their 3D career to understanding and deconstructing light

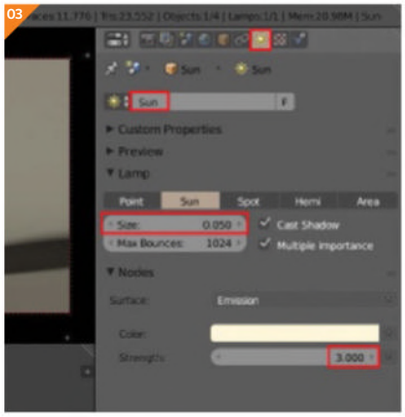
02 Material and shader setup In the meantime, we won't be using any fancy materials and textures, just so you can clearly absorb the idea and not get distracted by other prevailing elements. Add a grey (RGB = 0.5) Diffuse material to the sphere and backdrop. You can either use the Properties Editor or the Material Node Editor when adding materials to an object, the latter providing you with more control over the nodes as needed. At this point, make sure you have the Cycles render selected as the scene's renderer.

03 Key lighting setup In the following steps, we'll gradually build a lighting setup that is most commonly used in photography studios and product visualisations. The key light serves as the primary source of light in your scene. Its purposes include (but are not limited to) the following: highlighting/emphasising the main subject, providing the initial shape and adding the main shadow. Add a Sun Lamp, position it at an approximately 45-degree angle (Top View and Front View). Reduce the size to create sharper shadows and use a relatively higher lamp strength than the default value.

04 Fill lighting setup The fill light is used to add illumination to the unlit portions of your subject and to even out the shading. Usually, in photography studios, they use a softbox – a type of modifier that scatters and softens the light shot from a strobe or flash. In most 3D applications, we can replicate this by creating an object that will emit light. In Blender, add a Plane, position it adjacent to the Sun Lamp but on the opposite side of the sphere, scale it to a relatively large size, and add an Emission shader to it, increasing the strength as needed. You'll now notice that it killed the dark shading that we previously had, as well as adding a soft shadow opposite the emitter's location.

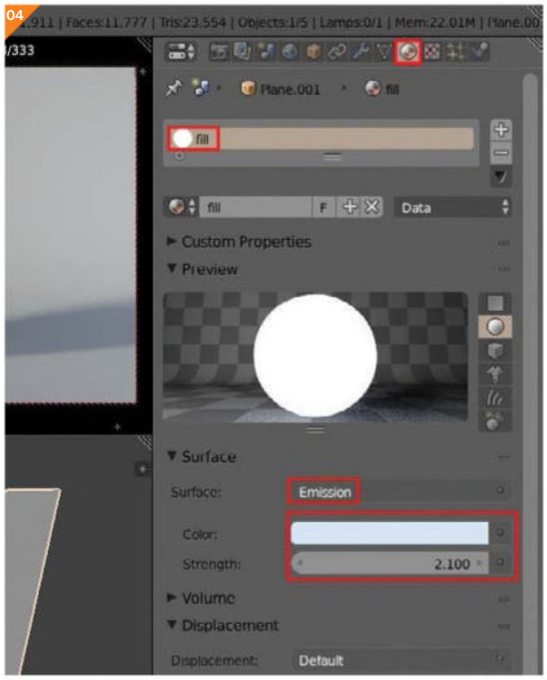


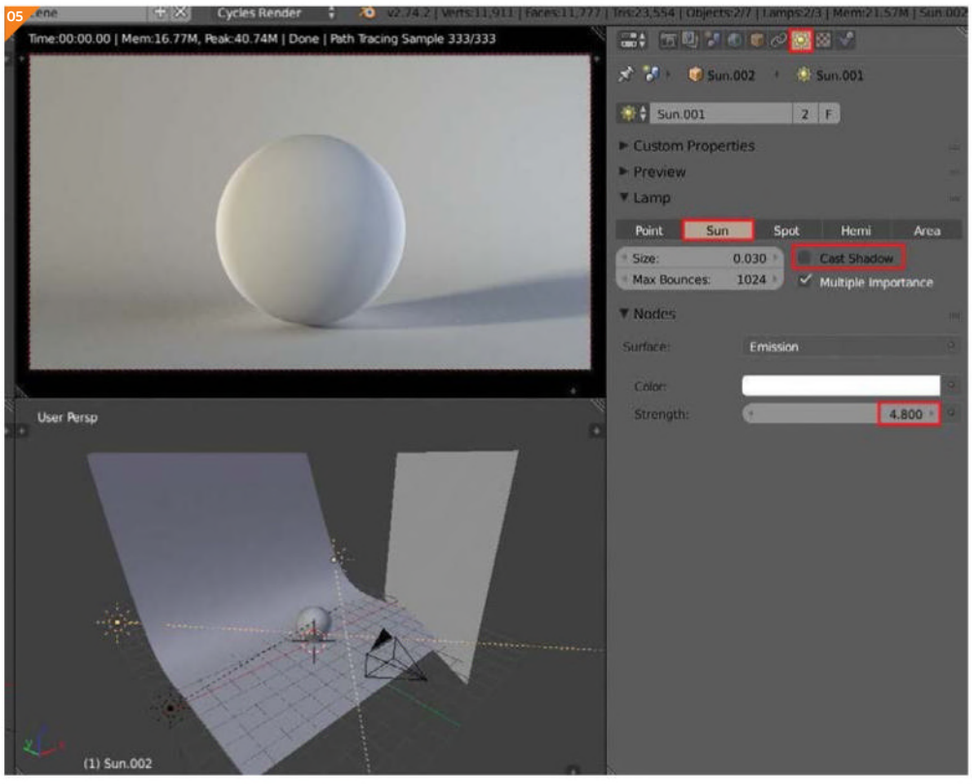
05 Back lighting setup As you may have already noticed, this completes the three-point lighting setup. This type of lighting is a cliché to artists, however, it serves its purpose well and lives up to its reputation too. Used sparingly, these lights will give life to your scenes that you would never have achieved otherwise. The back light, or rim light, is used to add definition to the overall shape and silhouette of the subject. When used on character portraits, it creates this wonderful scattering effect that is most noticeable on the hair strands and on skin. Add a Sun Lamp in Blender, position it opposite the key Light or the fill Light, and adjust the strength accordingly. Depending on your scene setup, you also may want to create another back light to compensate the other side of your model, as well as disabling the shadows being cast from these light sources.

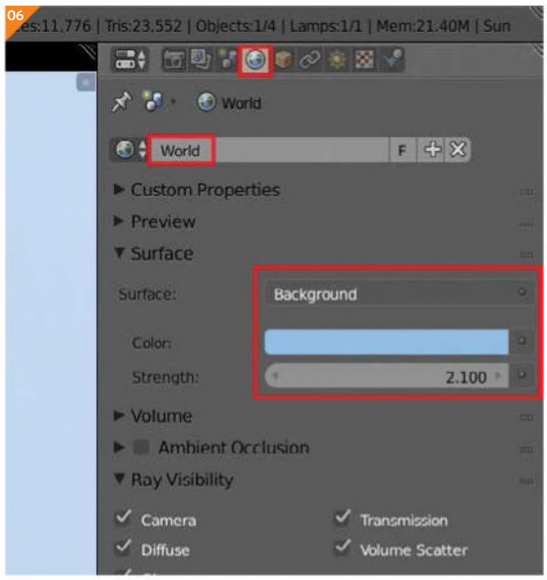


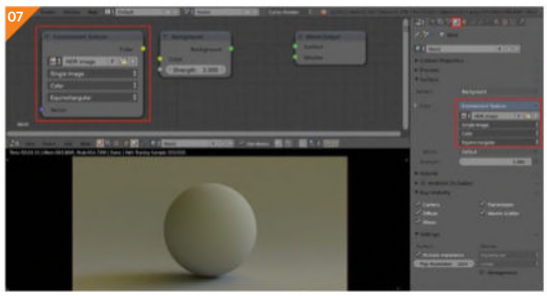
Lamp size matters

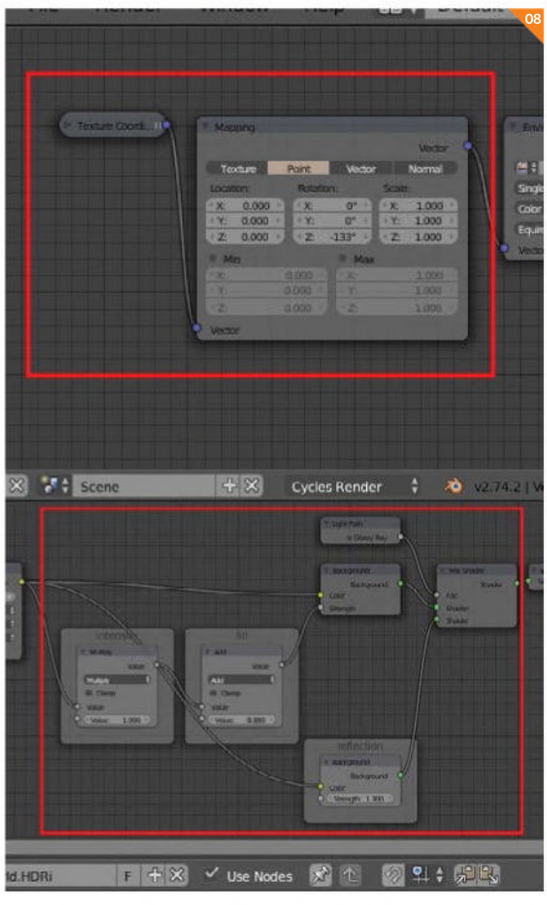
As with real life and in physics, the size of your light source dictates how sharp or soft your shadows are. The smaller your light source, the sharper and more defined your shadows will be. The larger your light source, the softer and more distributed your shadows will be. Keep this in mind when adding mood and adding drama to your scenes.

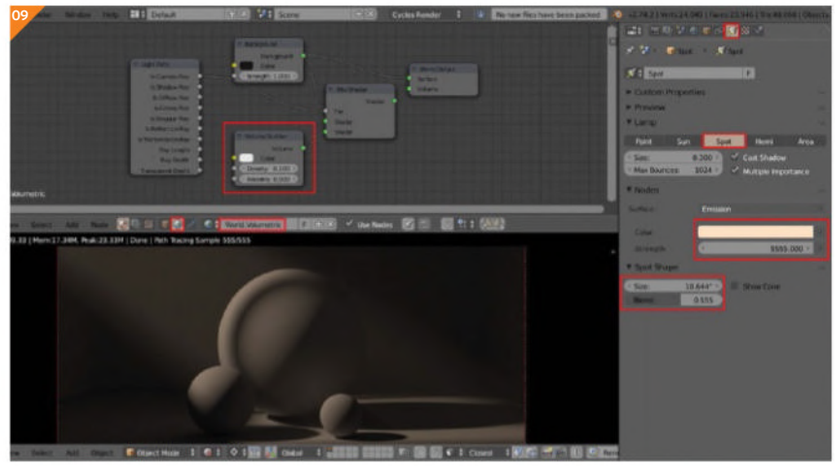


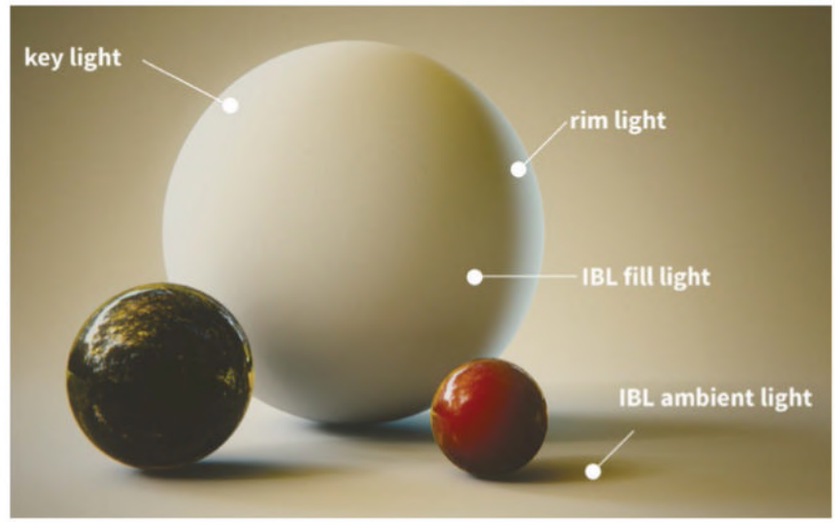












06 Ambient lighting setup The primary purpose of ambient light is to simulate the illumination that we receive from the sky, for example, on a cloudy and overcast day. When used alongside a CG and live action composite, this can help blend the CG scene seamlessly with real-life footage. Blender doesn't have an explicit way of achieving GI (global illumination), but by adjusting the World's settings, the results are almost convincingly similar. You can use this technique in conjunction with the key light method, but a better option is available on the next step.

07 Image-based lighting (IBL) One of the reasons we use HDRIs is because they have a higher bit-depth and can store additional lighting information that can dictate how your scene will be illuminated based on the image's light intensity values. This technique is often used by animation studios worldwide to give more life to their scenes as well as seamlessly blend several elements together (ie by shooting a panoramic HDR of the set). In Blender, you can do this by adding an Environment Texture and adjusting the strength accordingly.

08 Image-based lighting controller With the setup as seen in Step 7, we can further control how the lighting behaves by adding a couple of nodes in Blender that will enable us to rotate the environment texture, control the light intensity, control the light fill and adjust how much reflection it casts on reflective materials and objects. For the purposes of illustration, we've added two reflective (metal and dielectric) objects. However, there are still a lot of possibilities with IBL in Blender and we are only touching the surface with this tutorial.

09 Volumetric lighting setup And to conclude this brief overview to lighting, we'll introduce you to one of Blender and Cycles' new features. Volumetric lighting simulates situations which include fog, god rays, dust particles and the like. To achieve this in Blender, simply attach a Volume Scatter node into the World Output node's Volume Input, using a Spot Lamp as the primary source of light. By default, volumetric rendering in Blender is relatively slow, however, there are a few tricks to work around this with little or no detriment to quality.

REYNANTE M MARTINEZ
Empty, 2015

Software
Blender, Cycles

Learn how to
- Block out the scene and its overall composition
- Create believable shaders using Cycles nodes
- Develop a harmonious and minimal lighting scheme
- Add dust effects using image-textured planes
- Apply postprocessing effects using Compositing nodes

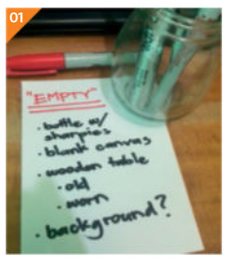
Concept
This piece, Empty, is a visual portrayal of one of the most difficult stages that artists and creators go through. And having gone through this a lot, I wanted to share the emotion that goes with it.

YOUR FREE DOWNLOADS
from **filesilo.co.uk/bks-798**
- Scene files
- Video tutorial
- Tutorial screenshots

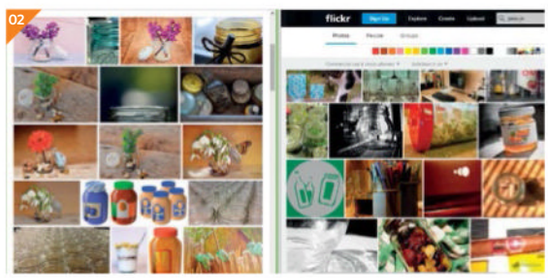
Render still life with Blender and Cycles

Create a visually compelling story using realistic textures and image-based lighting, all with simplicity in mind

This tutorial will guide you through the entire creation of artwork using Blender, starting from the blocking all the way through to the postprocessing phase. We will be guiding you through the creative process that is involved in creating a thought-provoking piece of artwork, while keeping the technical elements as simple as possible. The scene itself is rudimentary, however some of the shaders and textures are a tad more complex than they seem, and we will be guiding you through the entire material creation including some shader tweaks and scene optimisations for achieving fast and noise-free renders using the Cycles rendering engine. You'll also see how dust effects can be added using Alpha-masked hand-drawn textures and some modifier tricks that will add more realism to your models.

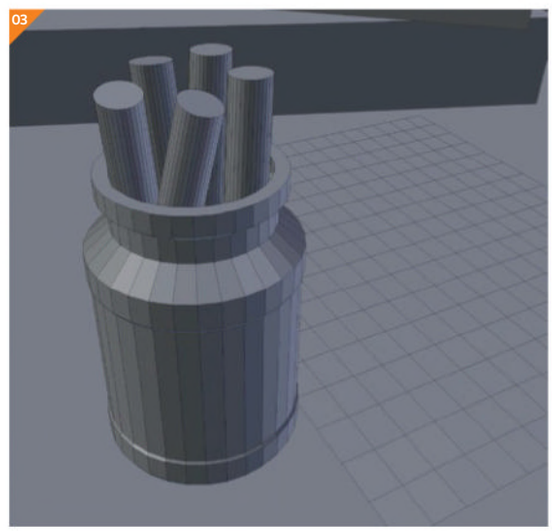


01 Plan ahead Before diving in on a project, make sure that the concept and story is as solid as it can get. There are usually two ways to do this: sketch the concept on paper and/or write a note about the details and idea of the concept – this ensures that you have laid down the initial foundation and will give you a broad sense of what you're about to create. For this project, we experimented with not having a sketch to start with, but rather having a few notes to make the idea more concrete and feasible. It also helped to have an actual tangible reference – luckily, we had one on our desk. If you can come up with a title to work on as early as possible, the better, and most of the elements you'll be adding later will revolve around this title. Don't hold back in the planning stage, the better you've planned ahead, the smoother your project creation will be – too much planning can ruin it too.



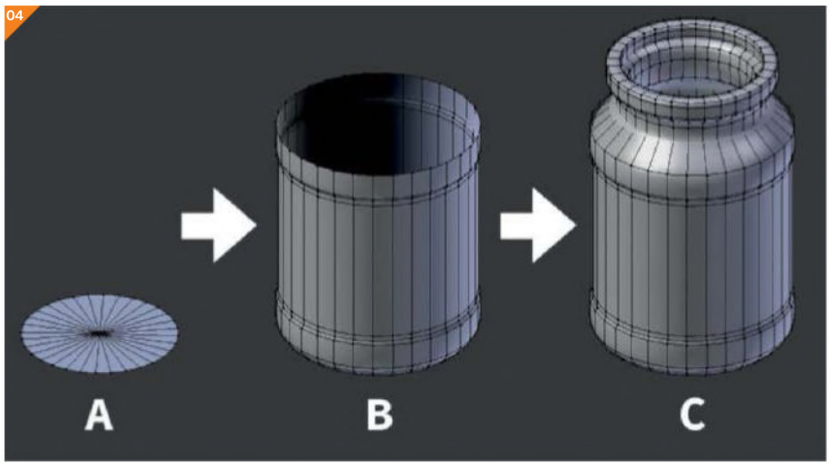
02 Build your references This is the stage where you put your researching skills to the test. Exhaust all your available resources: Google search, actual photos, tangible objects, sketches and so on. This will give you a better understanding of how certain materials and textures behave, especially if you couldn't have found them elsewhere. It will also give you more options to choose from in terms of colours and details. It strengthens the concept and idea that you already have, exposing other aspects that you might have missed.

Make a connection
Regardless of what genre you are working on and what style it has, make it believable as much as possible – this ensures that your viewers can relate to your piece and not just see a bunch of pixels on screen screaming colours and values. Create a visual harmony with the elements of your scene that will reinforce the message and story you're trying to convey. At the end of the day, it's the message that is most powerful in your artwork.



03 Block the scene Once your goals are clear, it's time to actually start the creation process. This is one of the most daunting stages of a project and skipping it can demotivate you. However, once you're aware of this fact, it's easy to counter it. No matter how complex your model and scene will be, breaking it down into manageable bits and pieces gives you more control over the resulting output. The best way to do this is to use primitives like cubes, spheres and cylinders to average the scale of the scene as well as their relative relationship with one another. You can then decide if one angle or aspect of your composition is working or not, before spending time on the laborious details later on.

"The better you've planned ahead, the smoother your project creation will be"

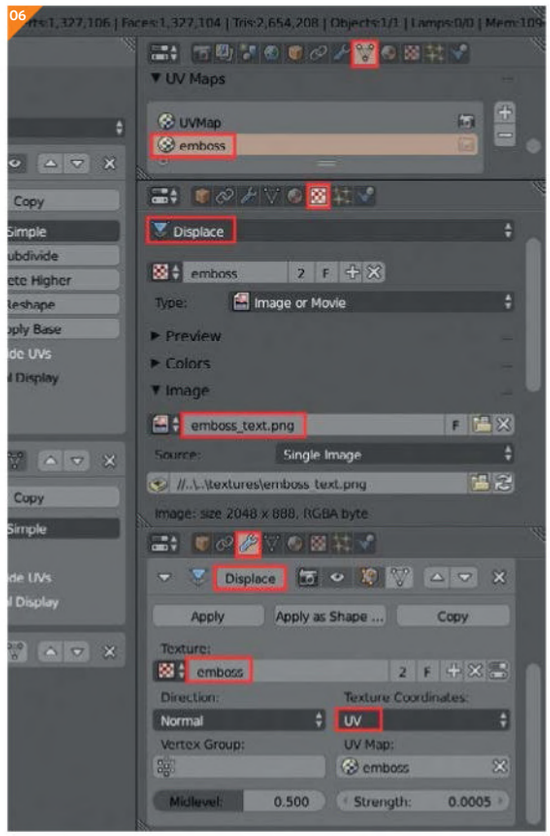


04 Model the glass jar To create the main subject of the scene, we'll use traditional poly modelling techniques. Start off with a circle with 32 vertices triangulated and merged in the middle. Select the outer edge and perform several extrusions and scaling. Continue with this procedure until you have completed the base shape with thickness. Finally, apply a Subsurf Modifier to smooth the object. Keeping the topology as simple as possible enables us to have more control over the form and will ease the process of texturing later. This same principle is also used in modelling the cup, saucer and books in the background.

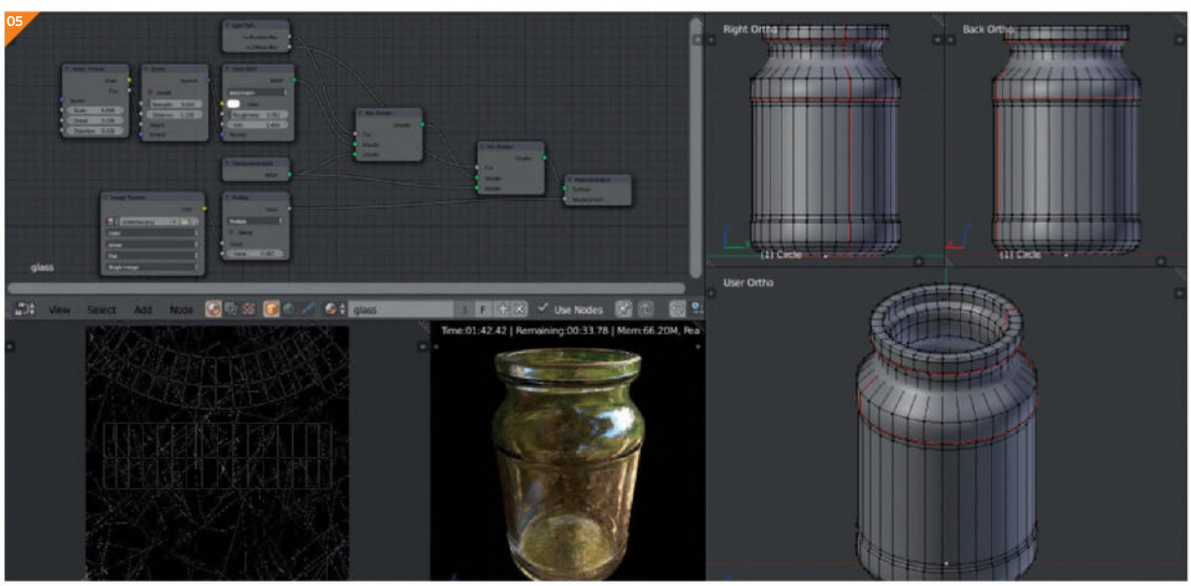
05 Texture the glass jar Since the glass jar takes up most of the visual information of the scene, it is necessary to add more realism to it by adding convincing textures. Create a simple UV map of the model and assign a scratch texture, which will indicate slight wear and tear, but keep it subtle. You can also add a Noise Texture and use it as a Normal map for the shaders, which is basically a mix of the Glass and Transparent Shaders. Doing this creates the illusion of depth and breaks the uniformity of the glass refraction.

Simplify as much as possible

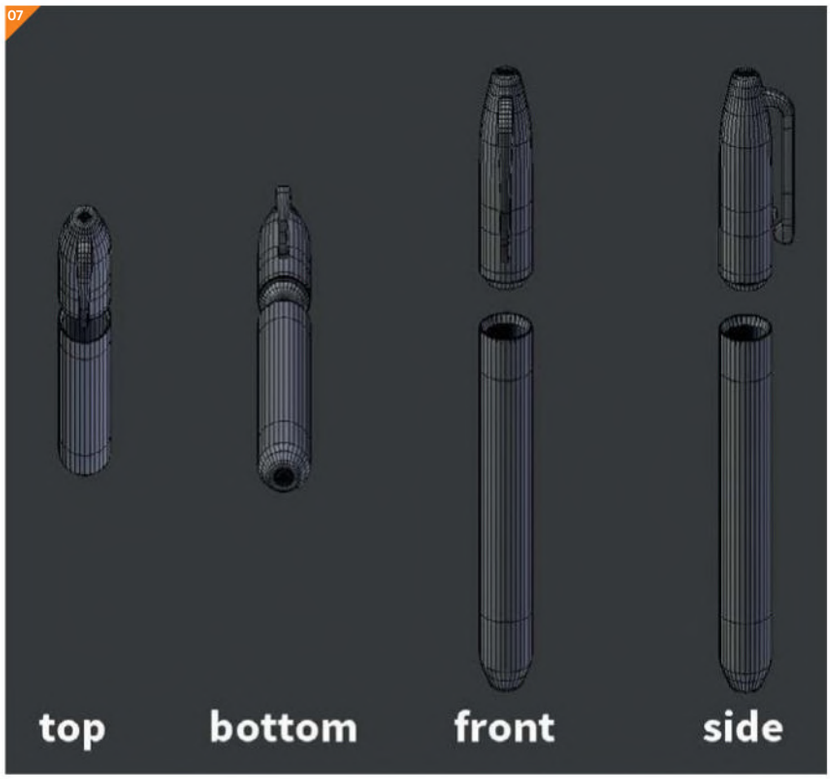
Often as artists, we are tempted to add extra details that wouldn't otherwise be seen. This is good if you have all the time you need, but in most scenarios, it's best to keep the details in moderation where needed and emphasise the necessary complexities in the main subject of your shot or scene. Not only is this beneficial to your time as an artist, which means you can work on more important things, but this is also technically more efficient, saving your computer the computation needed to calculate extra details (poly, texture sizes and so on). Work smarter, not harder.



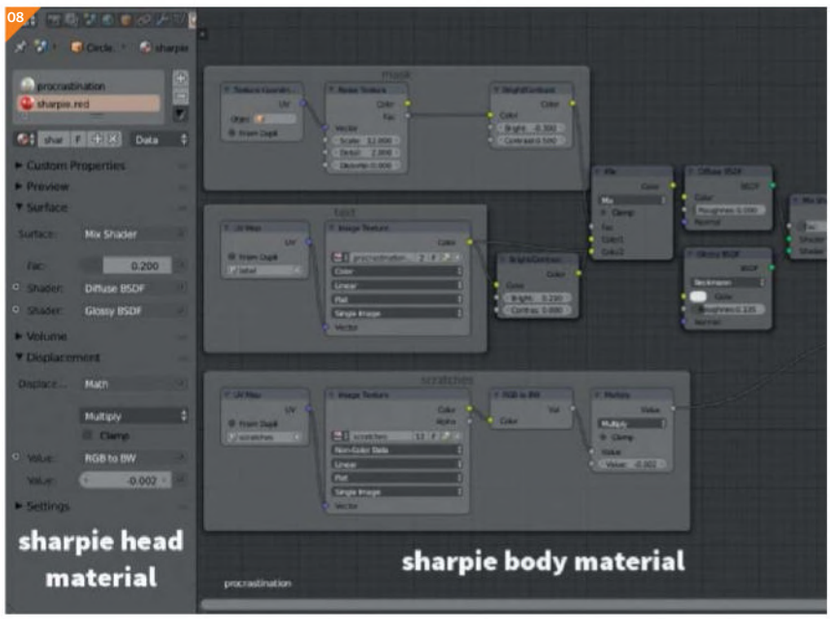
06 Create the displacements The text on the bottle might not be noticeable but it is this kind of subtle detail that adds believability. Using image textures as shader displacement might work well; however, doing the mesh displacement method looks more convincing in terms of how light behaves when it strikes the object's surface – at the cost of more polygons and memory. To do this in Blender, create a separate UV map, add a Multires Modifier to ramp up the amount of subdivisions, as well as adding a Subsurf Modifier to further pronounce the density, and finally using the Displace Modifier with the image texture assigned to create an embossed effect.



"Using a very simple Diffuse and Glossy Shader mix, you can achieve a fairly realistic leaf"



07 Model the Sharpies Using the same technique described in Step 4, we'll build the Sharpies starting off with a filled circle primitive, while keeping the scale relative to the glass jar. Duplicate the finished model and position them manually. There have been several attempts to try to automate the process via rigid body physics but it ended up being too chaotic and unpredictable; doing it manually will give you more creative freedom as to their composition and arrangement.



08 Texture the Sharpies To achieve the colour variation for the Sharpies, a separate material is assigned for the head portion. The same is true for the body, which contains different texts used as image textures. Doing it this way makes it easier to tweak the individual material settings however you wish. The Sharpie heads are a simple combination of Diffuse and Glossy Shaders with scratch displacements. The body, however, is a little more advanced, utilising several nodes to get the unevenness of the surface and texture.

Showcase

Reynante M Martinez

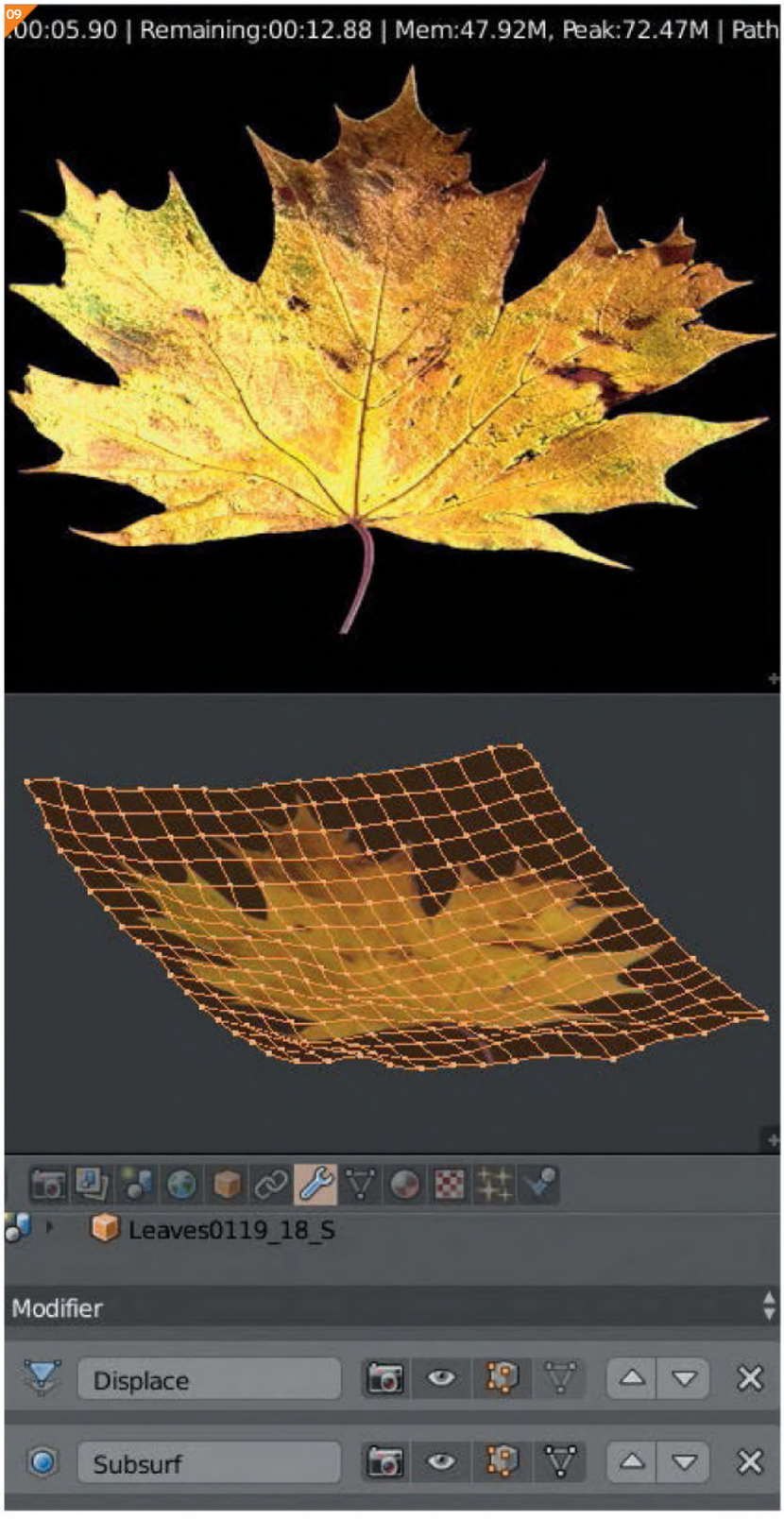
A self-taught artist and a Blender user for 11 years, Reynante specialises in visual storytelling and lighting. He has a background in graphic design, photography and illustration to name a few, with industry experiences that include academic teaching, art direction for short films and architectural visualisation.

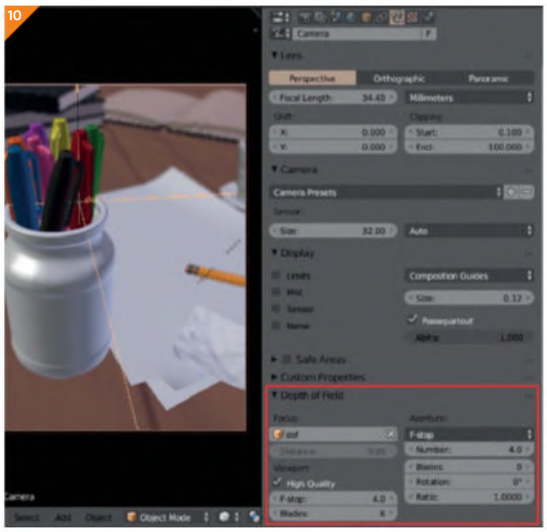
Rock On **Blender, Cycles (2015)**
This is my tribute to the scorching hot season that occurs in my country (Phillipines).

Fear **Blender, Cycles (2014)**
Inspired by a spontaneous idea that thoughtfully made its way into my consciousness on a random day.

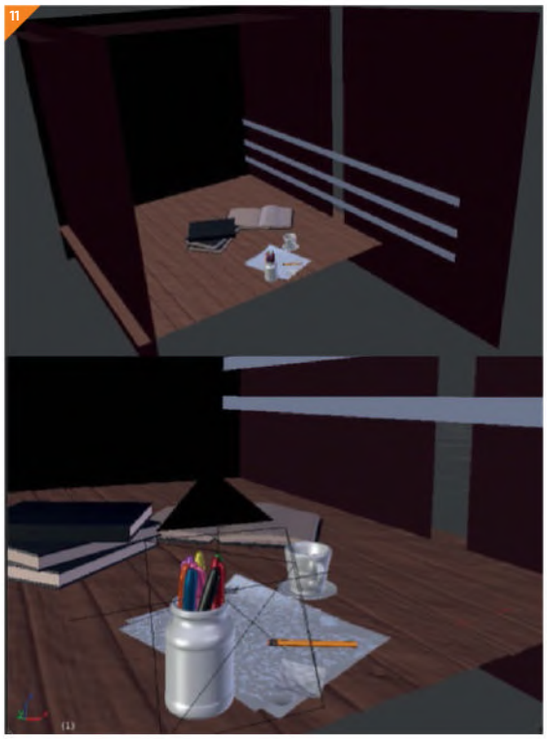
Playtime **Blender, Cycles (2015)**
Another doodle to express my odd imaginations and to detract from my usual routines for a while.

09 Add the leaf Using the 'Image as Planes' add-on in Blender, you can add image textures with transparency masks easily. But the real beauty is in tweaking the material settings. Using a very simple Diffuse and Glossy Shader mix, you can achieve a fairly realistic leaf without having to do any complex modelling. However, if the leaf would have been in focus, there might be a need to model it instead for more clarity and realism. This texture trick works for this scene since the leaf won't be much in focus.



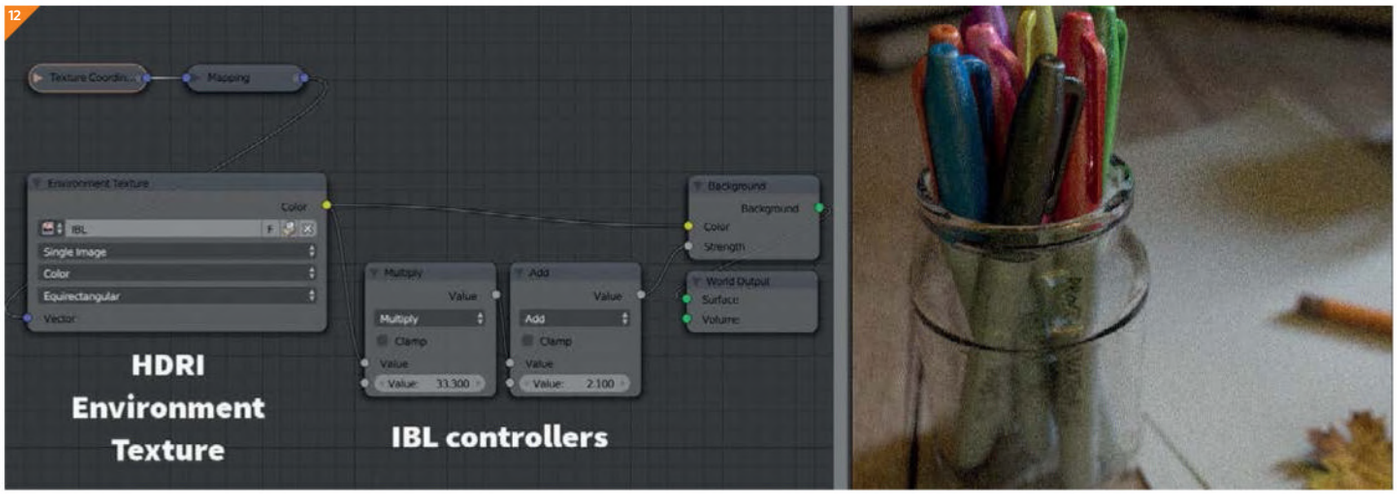


10 Set up the camera Once your models are set up, it's time to create a composition which will highlight and/or hide the elements in your scene. For some, this is one of the most crucial stages of the workflow process because composition plays a huge role in telling your story. In Blender, the Camera is just like any other object where you can apply rotations and translations as you wish. And to further emphasise your subject, you can use an Empty object as your 'Depth of Field' (DoF) focal point, while dialling the F-stop settings to control how shallow the blur effect is.

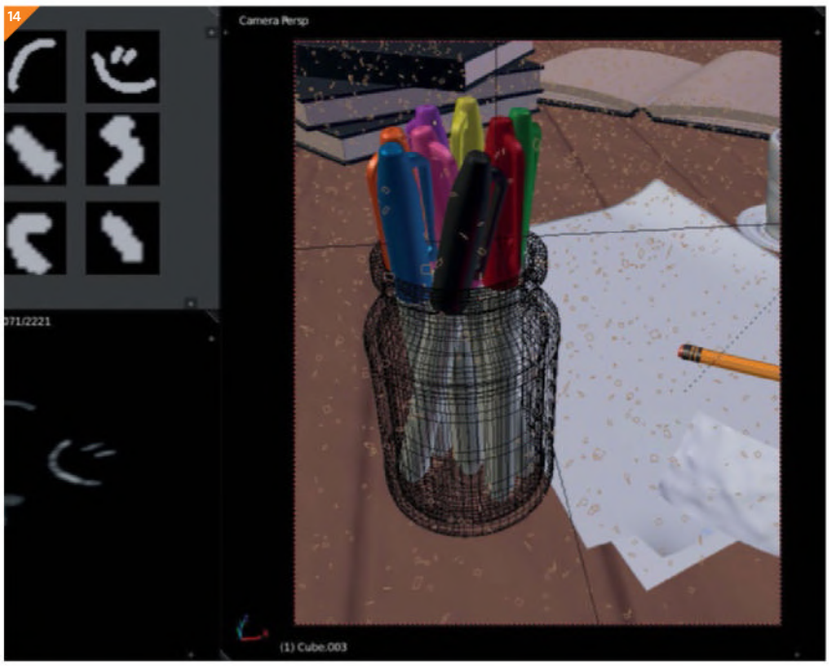


11 Fake the interior environment Often, it's not wise to entirely design the interior architecture of a project, especially when the focus is solely on an object where everything is blurred out. However, adding simple shapes to block out the basic form of the space is sufficient to create an enclosed space where light could bounce in as well as creating shadowed and unlit areas.

> Lighting plays a huge role in how believable your scenes will look

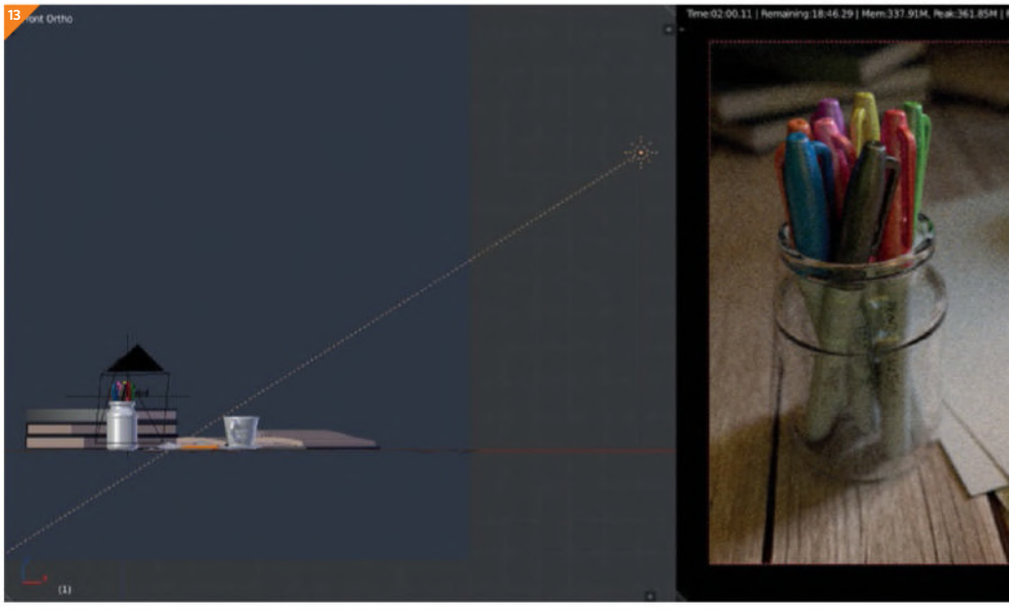


12 Image-based lighting (IBL) Using high dynamic range (HDR) images has been one of the most effective ways to illuminate a scene. Often, even with the absence of any lamp types. However, one downside to using them is that they take up too much memory, which is most evident when using large images with a high-bit depth. To balance this out, have an alternative lower-resolution image with a relatively higher bit depth than ordinary images like JPG or PNG. These IBLs also provide a great environment reflection on reflective objects. Via the node system in Blender, you can tweak and enhance the effects of IBL, where you can create more pronounced shadows and higher light intensities.



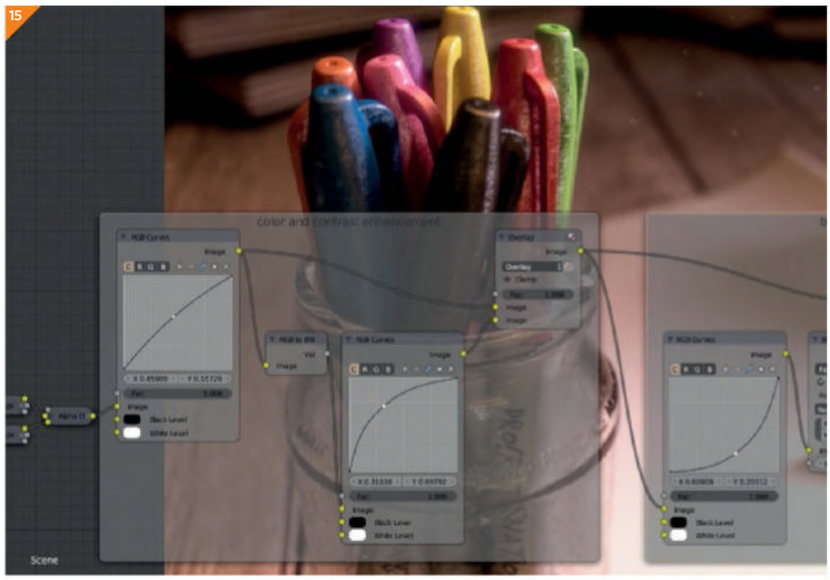
13 Additional lighting with Sun This step is optional. Adding extra light sources to your scene can make or break the intended visual composition. Always ask yourself what the purpose is to adding the lights. In this case, we are adding a Sun Lamp to further exaggerate the shadows and enhance the silhouette of the objects in the scene. To fully utilise this, use the lamp's size controls to modify how sharp or soft the shadows will be. You can disable any extra information like Diffuse, Specular and so on, and just keep shadow casting active if you wish to.

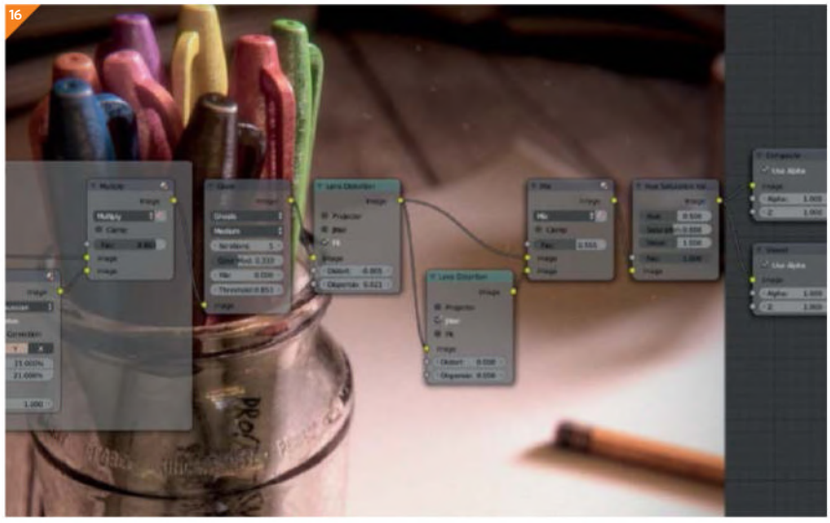
14 Create the dust Usually, dust effects are added in post-production after the image has been rendered, using tools like Photoshop or GIMP. However, adding them into the render gives you more realistic results like depth-of-field effects, volumetrics and the like. One way to do this is to draw random lines and use these images as Alpha masks mapped to planes, then using a particle system to distribute them across the entire scene.



Composition with lighting

Lighting plays a huge role on how believable your scenes will end up looking because well-composed light rigs can amplify the details of your objects and make them pop even more than before. Lighting can even add contrast to the layers of your composition so that we get a harmonious relationship between all of the objects in our image. Make sure that you make full use of settings like energy/strength, size and colour to further enhance the overall look of your renders.





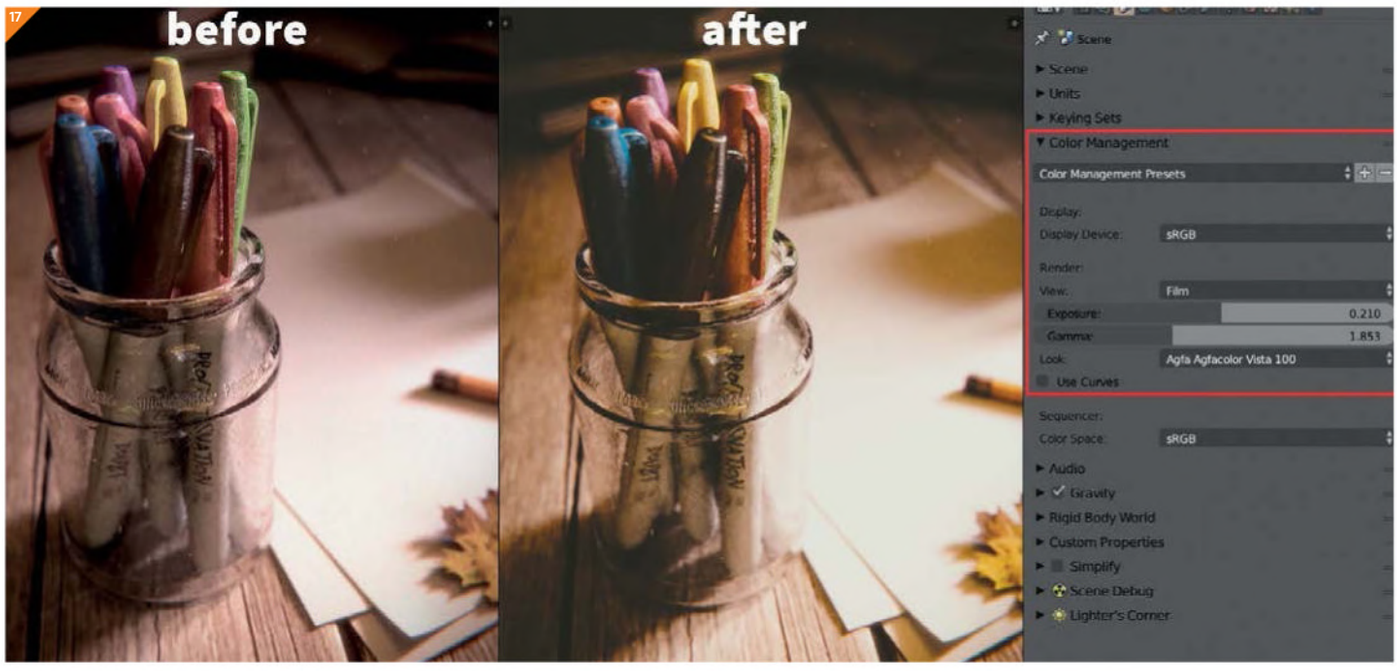
15 Colour and contrast enhancements Postprocessing has already been a part of major artistic workflow since the past couple of decades – and this is true for photography as well. The trick to doing this lies in moderation. Using Blender's Compositor is a great way of nondestructively enhancing the render, which works best when the raw render's format is a high-bit depth image. Using several nodes, we can enhance the colours and contrasts of the image thereby adding more visual quality.

16 Lens effects Imitating traditional lens effects can add another level of believability and realism to your renders. This breaks the computer-generated look that your raw renders have. It will also psychologically imply a natural phenomenon that is usually seen through traditional and digital cameras alike. Camera imperfections like vignette, chromatic aberrations and grains are among others that imply realism. In Blender, these are available via the nodes carefully placed one over the other.

17 Film emulation To finish the postprocessing effects on the image, apply a combination of film effects, preset looks and gamma adjustments. There are no hard rules to this and it's best to use your own visual taste and judgement. As mentioned in the previous steps, just make sure you keep it simple and subtle. In Blender, these settings are accessible under Scene tab>Color Management.

Animation postprocessing

Blender's Compositor is not only constrained to processing still images. Its true strength lies in animated sequences and/or RAW video files. Most of the settings (values, sliders, checkboxes) in each node can be keyframed according to any changes that have occurred in your animation and sequence. The processing of high-bit depth images like OpenEXR, HDR and Targa is also possible and is recommended when you want to perform any postprocessing operations.



Corona

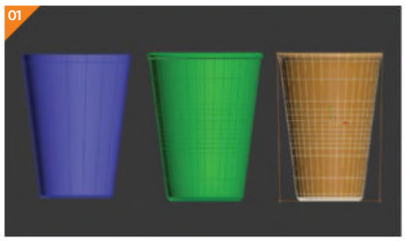
Peter Ang
dartofang.com

Peter has been doing 3D visualisations since 2008. His work has been featured in print and in art books

YOUR FREE DOWNLOADS
from **filesilo.co.uk/bks-798**
- Tutorial screenshots
- Video tutorial

Render realistic glass in Corona

To make these clear, crisp-looking glass cups, we will be doing some basic modelling in 3ds Max. Then we will sculpt it in ZBrush to achieve the shape and contours of the cup, and a little polypainting and unwrapping too. We will finish by exporting to 3ds Max. This method was chosen because it makes our modelling faster and more efficient. After modelling, we will then proceed in rendering using Corona. This renderer is very quick and easy to learn, and it gives you more realistic output in terms of material and lighting, with only a few tweaks in the settings. We will be starting the render from scratch so that you will understand some basic lighting and some different approaches to achieve a realistic render. After we finish with the render, we will do some postproduction in Photoshop and After Effects. A video tutorial and a 3ds Max scene will be included on FileSilo so that you can have a better understanding of how the scene was set up and how to use Corona in your own projects.



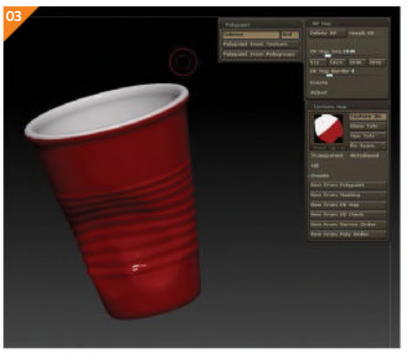
01 Model the crunch cup We will start the basic shape of the cup with a cylinder. From there we will add an editable poly and add little details to it by using Extrude, Chamfer, Shell and MeshSmooth. Remember to always use an Edit Poly Modifier every time you model, so that you don't need to do it from the beginning again if anything goes wrong with your model. We then use Swift Loop to connect edges and produce smooth edges when applying a MeshSmooth modifier. Take a look at the video tutorial to learn how to export and import properly from 3ds Max to ZBrush.

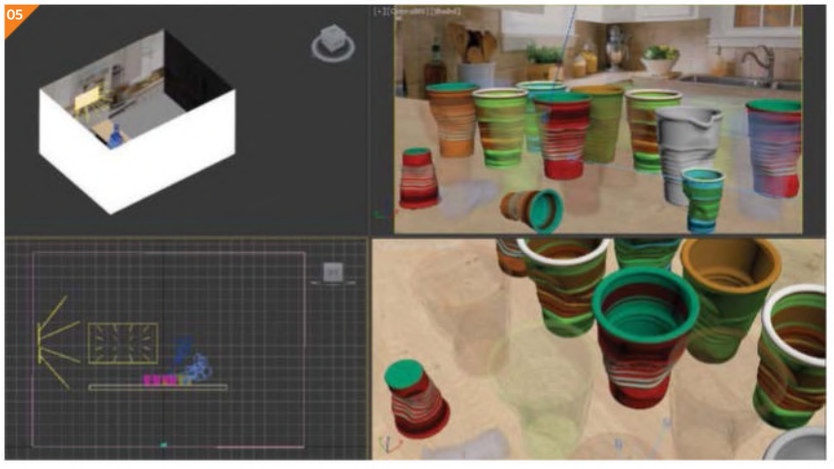
02 Sculpt in ZBrush We will start with masking the middle part with straight lines from the base. Invert the mask by going to the Mask menu and clicking Invert. After that apply Inflate under the Deformation menu and once we get the bulge that we wanted we will then apply Polish, which is also in the Deformation menu, to smooth all of the edges. Next we will be using the Move and SnakeHook brush to achieve the crumple of the cup. Using those brushes, slowly adjust the middle side of the cup to make crumples, based on our reference images.

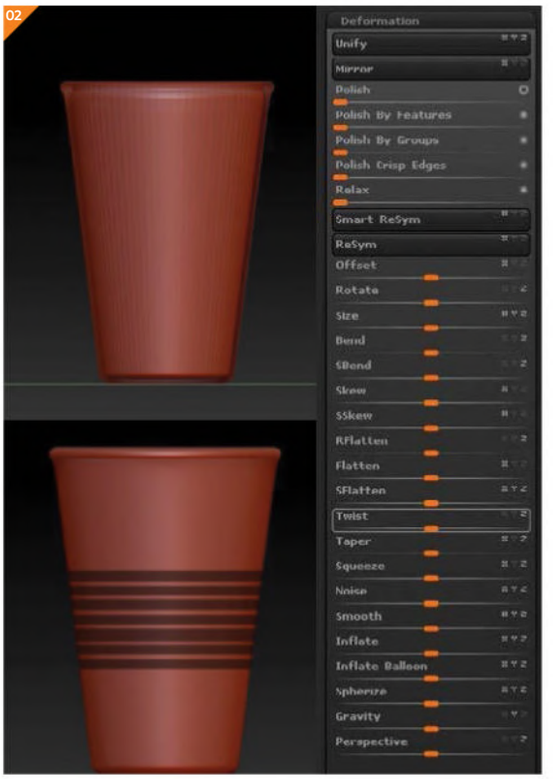
03 UV Master and polypainting After sculpting the model, we can apply the UV Master plugin. Before that, go back to our lowest subdivision, then we can apply UV Master under Zplugin. Click Unwrap all. Now when it's done, we can activate polypaint under the Polypainting menu. For this model we can use masking to create nice, sharp colours. After polypainting, go to the UV Map menu, and choose 2,048 resolution. Then go to the Create Texture Map menu and click Create Texture Map From Polypaint. Once it's created, clone it, go to the Texture menu, flip the texture map and export as a JPG.

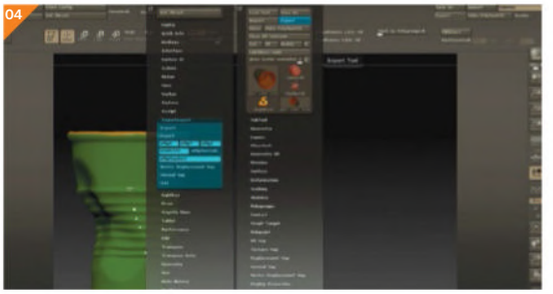
04 Export and import the model Usually when we export ZBrush objects, we use the lowest subdivision. But this time we will just use the third highest subdivision since it's just going to be a still image. So in ZBrush we go to Tools and click Export. Then go to 3ds Max, open File and choose Import.

05 Set up the scene First we will use a Standard Camera then set the right angle for the shot. Add a box with a wood texture for the base of our cups Then we will check the gamma in 3ds Max via Customize>Preferences>Gamma>LUT, and set this to 2.2. Now let's open the Render tab and select CoronaRenderer. For this scene the settings are pretty much all default since it works perfectly fine and optimises our render. If you hover your mouse to the values it will show the descriptions in it. We will be using a pt+pt under Corona's main settings and some WireColor and Z-Depth pass under Elements. A plane with CoronaLight material (with a kitchen texture map) was used to make the walls surrounding the glass model, this was also used as our background and reflection.







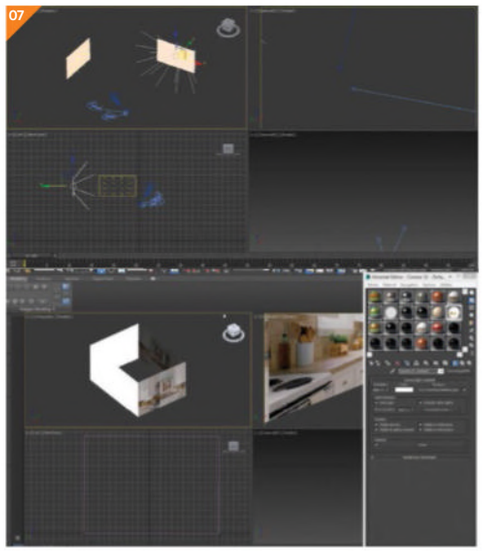


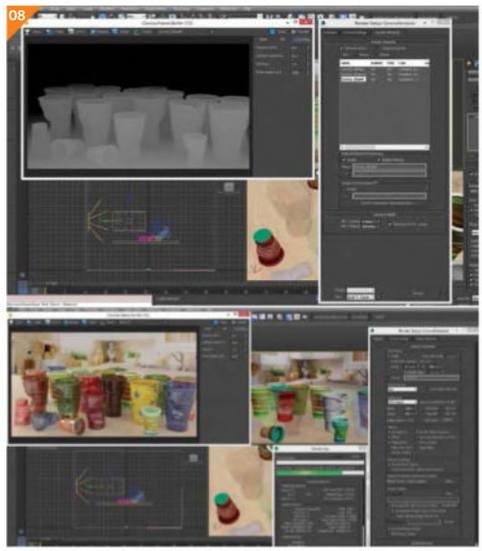
06 Texture the crunch cup For texturing the first cup, we will use the standard Corona material with Reflection set to 1.0, its Fresnel IOR set to 1.6, and its Glossiness set to 1.0. Then set Diffuse to the colour red. Next, make sure Refraction is set to 1.0 with a black colour, its IOR set to 1.5, and its Glossiness set to 1.0. With these materials we can now duplicate the cup on another slot and just change the Diffuse colour for the other different proxy cups. For the other cup with Unwrap UVs, we will load up the texture maps that we saved earlier on when we were in ZBrush. The settings now will be a little different from the first cups. So set Reflection to 1.0 with Corona Solid Tex as a white colour, then change its Fresnel to 2.0 and its Glossiness at 0.84. Move on to Refraction and set that to 1.0 with a black colour, with its IOR set to 1.6 and its Glossiness set to 1.0 After we set up the texture, we will then rotate it randomly to add variation in the shot.

07 CoronaLight materials There are different ways to light up a scene. But always keep in mind that you need to put your lights in the right position to reflect how a real interior light works. This is so that you don't have any weird lighting results on your image. For this, we will be using two Corona plane lights to be our source for the main lights, one on the left side with an intensity of 50 and one on the front with an intensity of 10 – these lights will be our windows. Now we will put a CoronaLight material on two planes with the multiplier of 5, and this will give us the reflection from our cups. The other two planes at the back will be our background with just a standard Corona material.

08 Test renders Every time you do a test render you should always practice rendering the elements, so that you can put the right amount of Z-Depth pass or Ambient Occlusion when you render the final high-res image. If you click the Render button, another window will show up showing the progress of the image. From here you can adjust the contrast of the render under ColorMap, play around with the settings, and also set the Exposure to 0.0, Highlight Compression to 50, Contrast to 1.0 and White balance to a number between 4,500 to 7,000. Once you're satisfied with the render you can now click the Cancel button and click Steady on the Save icon and click 'Save all' to save the image and its elements.

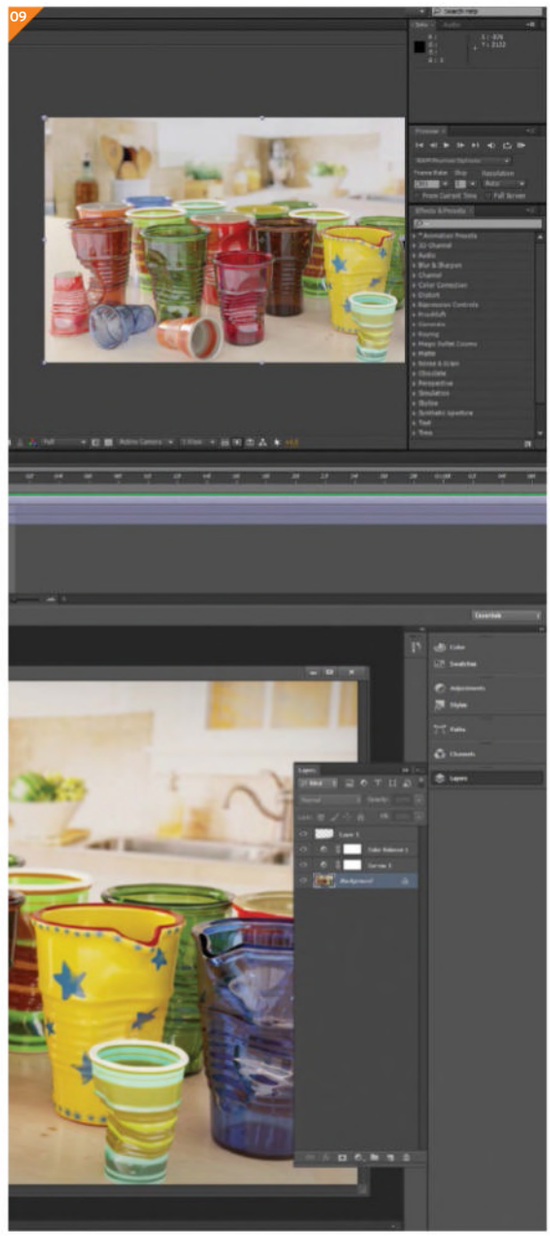
09 Postproduction in Photoshop and AE Before we do some editing in Photoshop, we will take the raw image and the Z-Depth in After Effects to apply some depth of field using the plugin Lenscare. Go to File>Open project and select the raw image and Z-Depth pass, then drag the images onto the empty Composition window one by one. Then go to Layer>New>Adjustment Layer. Add an effect on the Adjustment Layer by going to Effect>Frischluft. Under the effect controls in the Adjustment Layer, apply the Z-Depth that is in the depth layer with a radius of 10 or any value that you think you like. Once we're satisfied with the depth-of-field effect, we will go to Photoshop. For this image the colour correction is pretty much minimal, we will just add some adjustment layers with curves and a little contrast. Voila, we now have a photorealistic crunch cup!





Corona settings

The default Corona settings will work mostly in every scene. If you're rendering an exterior scene you can use pt+pt under Main Settings>Global illumination and pt+hd for interiors. You can also tweak the depth of field and scene cameras under Global Camera Settings, but usually I just enable the Depth of Field and change the blades for the bokeh.



Real-time dof vs Z-Depth

Take note that if you use real-time dof in Corona, the render time will be longer and you cannot change the depth of the image anymore in postproduction. But what if you render the image without the real-time dof and render with the right Z-Depth pass? You will get more control of the focus distance that you want on your image, especially when using Lenscare in After Effects.

STEFANO TSAI
No place to hide, 2015

Software

3ds Max, V-Ray, Photoshop

Learn how to

- Make a comic image with VRayToon rendering
- Add diversity to VRayToon outline widths
- Use Photoshop filters to make the image interesting
- Bring a comic-style effect to rendered scenes

Concept

Take a Lockheed P-38 Lighting as your reference for creating a lightweight patrol aircraft, which has distinctive twin booms and a central cockpit. It is used as a short distance patrol aircraft in the story and can carry two pilots maximum.

YOUR FREE DOWNLOADS from filesilo.co.uk/bks-798

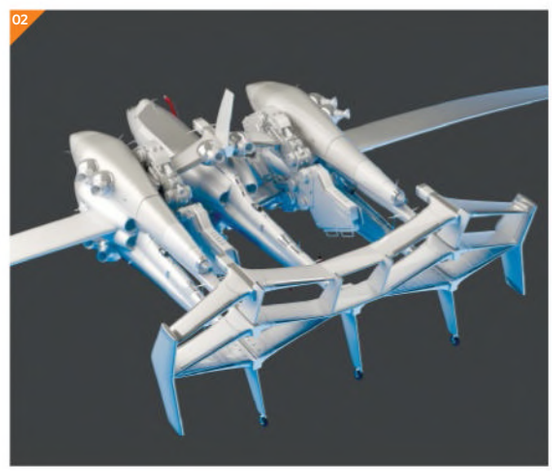
- Tutorial images

Create comic book style renders

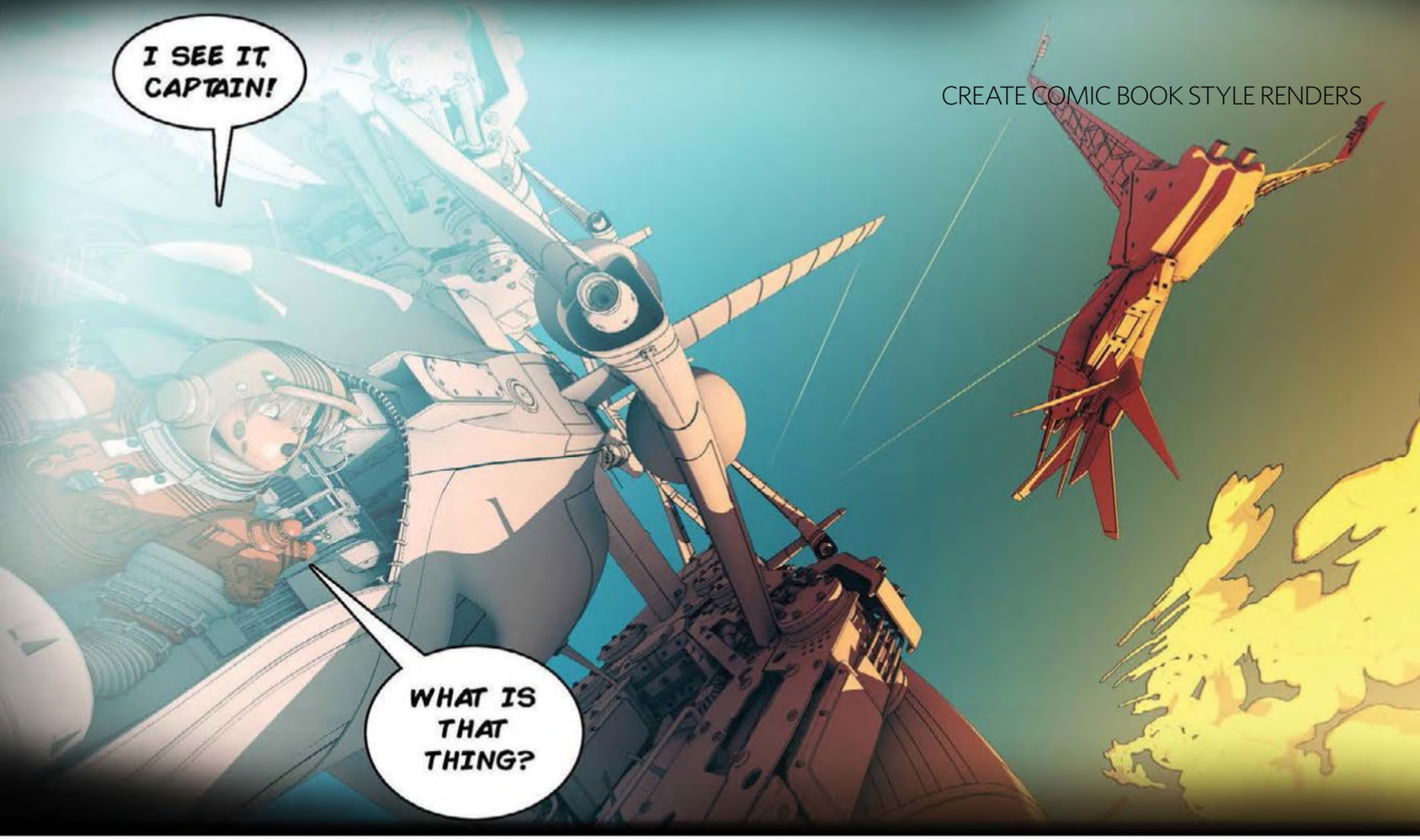
Use 3ds Max and VRayToon to visualise an aircraft concept and bring it into a virtual comic strip world

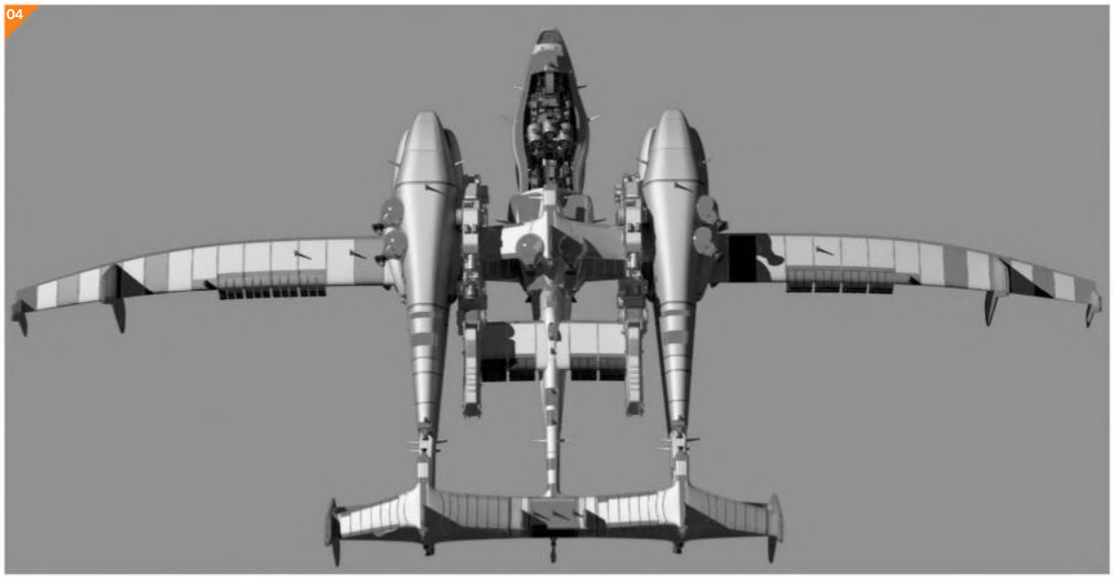
Throughout this tutorial, 3ds Max and VRayToon will be used to teach all of you how to render in a cool comic book style. We will use multiple VRayToon outline settings and also make the image much more interesting by adding layers with Photoshop filters. Without rendering, the difficult part is to find other interesting details for the image without losing the main focus. After all, you can't make a good image without a focal point, so here we have to use the minimal elements to create the desired excitement and vitality for the scene. The tutorial starts from the aircraft itself then extends into the environment surrounding the aircraft.

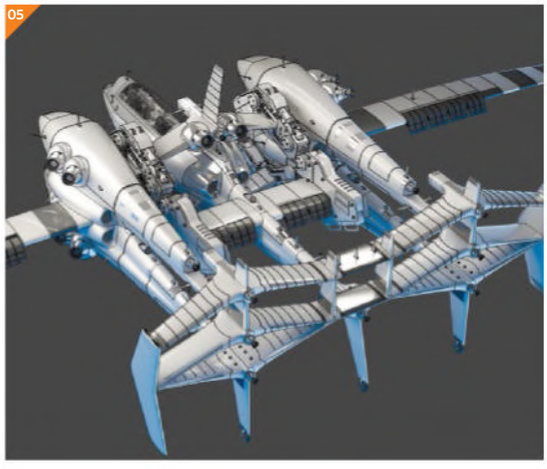
01 Start with an original concept The lightweight aircraft was created for the first chapter of the comic *BPI: Blue Patrol Industrial.* This is the vehicle for bringing the main character into the world. To make it more realistic, it contains a lot of functional elements for both the exterior and interior. The original concept has colours and a realistic V-Ray render. The challenge here is to bring this to a comic book style but still keep the excitement for these important assets.



02 Apply the simple grey materials It is so much easier to start with an empty sheet for a new style because it gives less distraction. It is a bit scary to start from nothing but it is definitely the most effective way to kick in. I took the concept models and applied a few grey default V-Ray materials. Some of them are reflective and some of them have a bit of a darker grey colour.





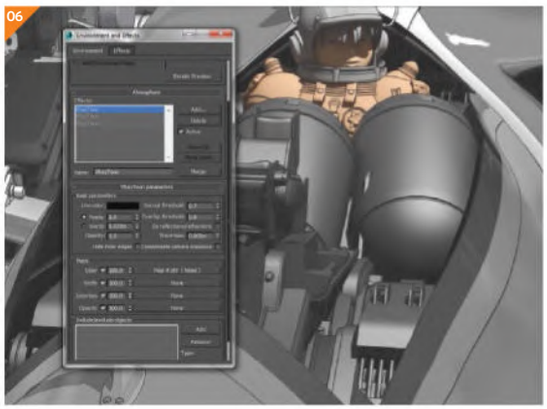


03 Make a few differently toned materials Now we use a range of greys, with some greys more intense than others. We also add an engineering shade of orange into the aircraft to bring in a bit of excitement. We only add in one more colour here; if we make it too complicated it will kill the simplicity of the comic style we want to achieve.

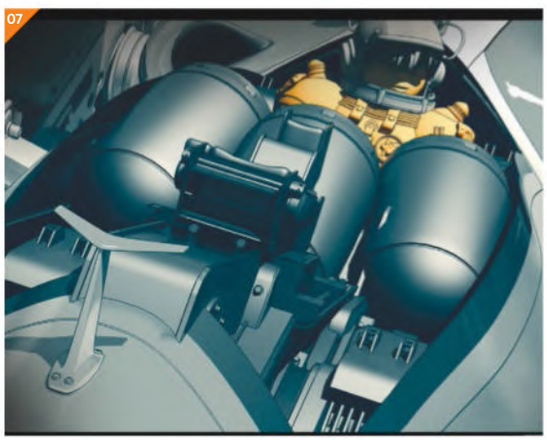
04 Start applying multimaterials We've applied six materials to the models, and the comics usually use only a few simple tones to completed images. Here we keep orange for the character suit as the character needs to be the camera focus, so it needs to have more visibility and to be the focal point. Using the remaining five colours, play around with different portions and combinations for the best balance.

05 Turn on VRayToon render With VRayToon render on, we now also have the outlines around the models. It gives the models the right kind of comic look. However, if we compare this to a cartoon production, the outlines here are a bit boring. This is because they all have the same width. If this is the only big object in the scene then that could be really dry for the viewer's eyes.

06 Play with line width To make it more interesting, we need to give it a different width for different objects. Create more than one setting for VRayToon, this way they can hold different settings for the different groups. For example, you can use less obvious lines for background elements and use stronger lines for the area that is nearer the focal point.

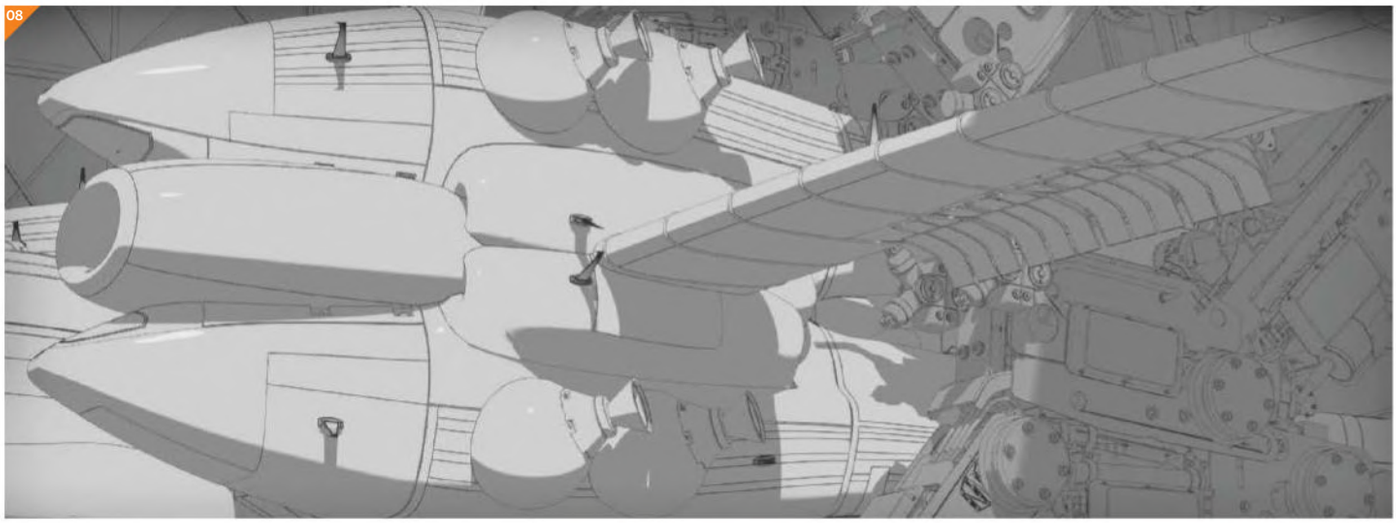


07 Play with strong lights With all the different line groups set up, we now have more visual variety. The background elements were pushed back and the focal point now lands on the character area. However, if there is no light there is no life. Lighting does play a very big part in a good image. Here we placed a strong sunlight and made sure the edge of the shadow lay on the character's face, so there will be a main focal point for this image.



Liven up the scene

After setting up a basic VRayToon, we now have the beautiful toon shader lines around the 3D models. we started to focus on how to bring life into the scene. To make it feel more humanistic, we can add more human details into it that can be dirt, noise, graffiti and so on. The strongest element you can use is lighting. Good lighting can give a scene new life or even bring it some drama that can help storytelling and build up the character's personality. Here we are going to use a bit of 3D software help to get an AO pass. Then we use Photoshop filters to help make different focal points across different cases.



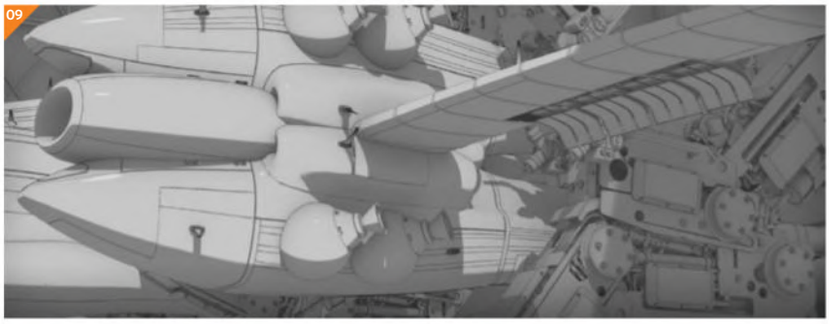
08 Take a comic scene as an example Here we took a comic scene to show how we turned it into a comic-style page. This is an aircraft launch scene, the patrol aircraft is going to be launched through the catapult. First, use the same VRayToon outline setting from Step 1 to apply to all elements and make sure the lighting has been put into the best place so it can cast a good shadow and make a good focal point. The area near the middle left in our image is the focus.

09 Add AO This vehicle is for a comic style, we should keep it flat looking. However, since we made all of this in 3D, there is no reason to not take advantage of 3D software. We took the same models and rendered out their AO, known as ambient occlusion. If you are a V-Ray user, there is a handy script called Vray Ambient Occlusion by populate3d.com.

10 Add details with Photoshop filters After applying an AO render in Photoshop's layer with multiply, start to add more dirt and noise details into the scene. To make it more interesting, apply more Photoshop filters – one on the left side of the image to make the lighting stronger and another one on the right-hand side of the image to add a red tone to create more contrast in the image.

Simplicity is best

I really like the movies created by Mamoru Oshii, Katsuhiro Otomo and Hayao Miyazaki. They have the true blood of Japanese animation – a visual feast of mechanical design. The concepts in these movies are just so inspirational for artists. However, when you have started learning from their work, you should find that the truth behind all of the complicated mechanical and set design is 'simplicity'. That is, the magic of cartoon and comic. No matter how complex those design are they will be united by art director eyes and harmonised by colours, lighting, mood – most importantly the final result is always so easy to watch and understand so that you feel the simplicity, energy and love put in.



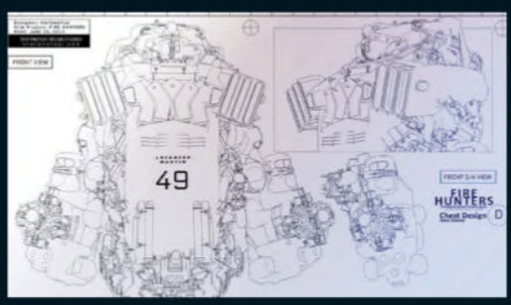
Showcase

Stefano Tsai

Stefano is a concept artist, a designer, a dream machine and virtual environment builder living in the UK. He loves everything about details and has dedicated his life to this since he was a child. He enjoys the process of bringing concepts to life.

***FireHunter Set Concept*, Photoshop, 3ds Max (2015)**
A concept for the short animation 'Fire Hunters'. This is the concept for a lower street set with a lot of steam, machinery and advertisement boards going on and also some futuristic vehicles mixed with old, classic vehicles.

***FireHunter Mech Cockpit Interior*, Photoshop, 3ds Max, (2015)**
Another concept for the short animation 'Fire Hunters' – the concept for the cockpit of the mech. After trying a few versions of it, this is the one that I love the most.



***FireHunter Mech Cockpit Exterior*, Photoshop, 3dmax, (2015)**
This is an exterior mech concept for the short animation 'Fire Hunters'. It is the chest section, but you also can see the arms. This is the mech for the main character.

11 Make the interior shot The space is quite narrow for placing a camera for a special cabin scene. To make it work, you can either create a wide-angle camera or create a cross-section scene for the shot. We've chosen the latter and cut some panels off to make the shot possible. This also added more details for the scene and helps to show the depth of the story and believability of the aircraft.

12 Make a medium shot in the clouds For a medium shot, make the VRayToon lines much lighter and thinner. You can also add more environment colours into layers to help the aircraft to blend into the scene. To make it interesting, add a strong light on the left side to ease some attention and on the right side add a red tone colour filter to push the eye's focus to the centre of the image. We made the background terrain much darker so that it will not fight with the aircraft.

13 Make a distance shot For a distance shot, sometimes you need to turn off VRayToon render to avoid the line noise on small objects. Here the aircraft becomes small, almost like a dot. If we apply VRayToon to the aircraft, it will become too noisy and jump right to the front. However, we need it to be pushed to the background, so in this case, we turned off its toon shader, but the rest of the elements are kept the same. To create less interference, you can use a pure black material for some ground level elements, so that again the focus now lies with the aircraft.

Simulate macro photography

Creating a truly realistic macro photography render requires a physically based renderer, some knowledge of basic photography, and patience. Octane is a GPU-based renderer that is gaining popularity because of how it balances ease of use, speed, and overall rendering power. Octane is a stand-alone rendering application that also offers additional plugins for a variety of the most popular 3D applications.

This tutorial goes through some techniques developed using Octane for Maya to create the macro photography renders. To use Octane you must purchase the stand-alone version and then also purchase the plugin. The stand-alone application is an easy-to-use node-based system with a GUI that is reminiscent of a compositing program such as NUKE.

This tutorial focuses on the Maya plugin which integrates perfectly and feels more stable than Maya's own mental ray renderer! Both the stand-alone version and the plugin are very affordable, a joy to use, and produce results that can be superior to mental ray and V-Ray. The rendered images you see in this tutorial had little to no postprocessing in Photoshop beyond slight colour and exposure adjustments. The main advantage of using the Maya plugin versus the Octane stand-alone is that animations can be rendered without the need to export an FBX sequence to the stand-alone version of Octane. Octane requires a CUDA-enabled graphics card. For this tutorial we are using a PC running Windows 8.1 Pro, Maya 2015, with 64GBs of RAM and two Nvidia GeForce GTX 780 Ti graphics cards.

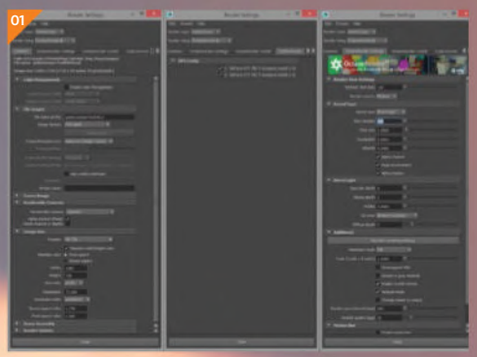
Maya, Octane

Eric Keller
bloopatone.com

Eric specialises in scientific visualisation and teaches at the Gnomon School of Visual Effects

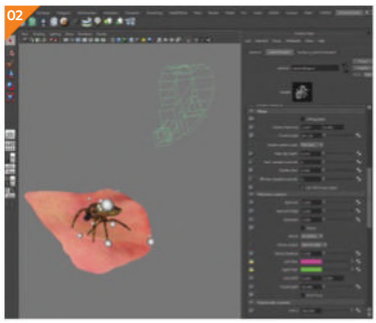
YOUR FREE DOWNLOADS
from **filesilo.co.uk/bks-798**

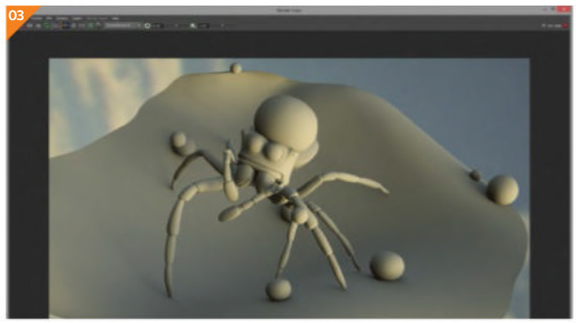
- Tutorial screenshots
- Source images
- Scene file
- Video tutorial

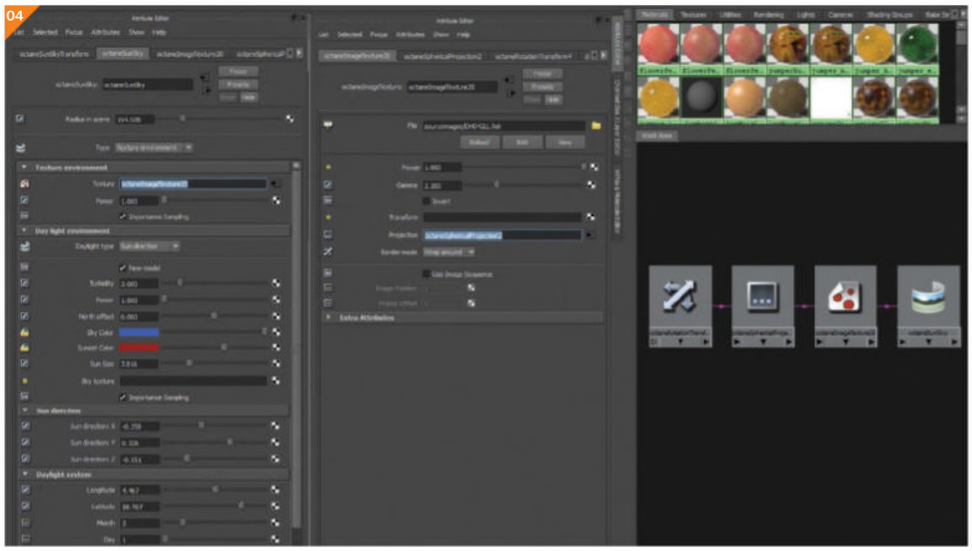


01 Configure your render settings

This tutorial assumes that you have both Octane and the Octane Plugin for Maya installed. Let's start with loading your scene. In the Common Tab, Maya's Render Settings window, set Render Using to OctaneRender. Choose your rendering camera and resolution – the HD 720 Preset works for this example. Switch to the Cuda Devices tab and activate your CUDA-enabled graphics card(s). In the OctaneRender Settings tab set the Kernel type to Direct Light and Max samples to 200. This is a low-quality preview setting which will keep interaction fast while you set up. You'll change these options for the final render later.







Paint Effects shader script

This simple MEL script can be used to apply an Octane material to a number of selected Paint Effects strokes.

```
//Get list of selected strokes
string $selStrokes[]=`ls-sl`;
select -cl;

//Prompt user for name of Octane Material

string $button = `promptDialog – title "Select Octane Material"
            – message
              "type the name of the Octane Material you want to apply to selected strokes"
            – button "OK"
            – button "Cancel"
            – defaultButton "OK"`;

string $octMatName = `promptDialog -q`;

//Loop through list of strokes and apply named material
for ($i=0; $i<`size $selStrokes`; $i++)
{
//loop through selected brushes and connect octaneMaterial
connectAttr -force ($octMatName+".outColor")
($selStrokes[$i] + ".octMaterial");
};
```

02 Set up a camera Choose the OctaneRender tab on the Maya shelf and click on the camera icon to add an Octane camera – this is just a Maya camera with additional Octane settings. Many of these settings override standard camera options when rendering in Octane. Position your camera to look at your subject. In this example, we're using an adorable jumping spider model created in ZBrush, perched on a flower petal. Select the camera and open the Attribute Editor. Set the Octane Camera type to Thin lens and set the Aperture to 0 so that the image is entirely in focus while you work.

03 Add a Sun/Sky node From the Octane shelf, add an Octane Sun/Sky Environment. In the Attribute Editor for octaneSunSky you can set the Type to Texture environment as an IBL texture, or set to Day light environment to use the procedural Octane physical sun – we prefer using Texture environment. Click on the icon next to Texture to load an HDR image. Pick Octane Image Texture from the Create Render Node window. Then, use the File field to select your HDR image. Click on the texture next to Projection and add a spherical projection node. In the Attribute Editor for SphericalProjection click on Octane Full Transform. You can use this node to position and rotate your IBL image.

04 View your render Open up the Render View window and choose Render>Snapshot>Camera 1 to set the test render camera, and now you'll see a wireframe of your scene. Click the IPR button to start a test render. It takes a moment to compile the geometry for the scene. Then you'll see the render appear and update. You can change lights, shaders, and the camera view and see the results. Occasionally you may need to press the IPR button to refresh the scene, this is usually when you need to move geometry or add a new object or shader. The statistics at the bottom show your progress. The render continues until the Max samples is reached. It will look grainy until you increase Max samples.

05 Load Octane Material Presets There's a database of user-created Octane materials accessible from within Maya. In the OctaneRender Settings tab of the Render Settings window turn on Enable LiveDB refresh (under Additional). Note: an internet connection is required. Switch to the Octane Live DB tab and click on the arrow to load the database. For the waterdrops in the scene, use the Water drops material in Organic>Liquids. Click on the Create node button to load and then select the material in Hypershade. Apply it to the geometry and press the IPR button in the Render View window to see the result.

06 Create custom materials

In Hypershade click on Octane materials. The material types here include Diffuse, Glossy, Specular and Mix. We will use Specular for transparent or translucent materials, Glossy for shiny materials, and Diffuse for dull or rough materials and light-emitting objects. Mix material combines two other materials, their respective strength is controlled using a value or a texture. We have mixed a Specular and Glossy material to create the translucent spider, and used Diffuse material applied to a polygon plane to create an area light. Add an EXR texture of a soft light box to the Diffuse material's Emission channel to create a realistic light. Texture, Displacement and Normal maps can add a lot to the realism of the materials.

07 Apply shaders to Paint Effects and hair

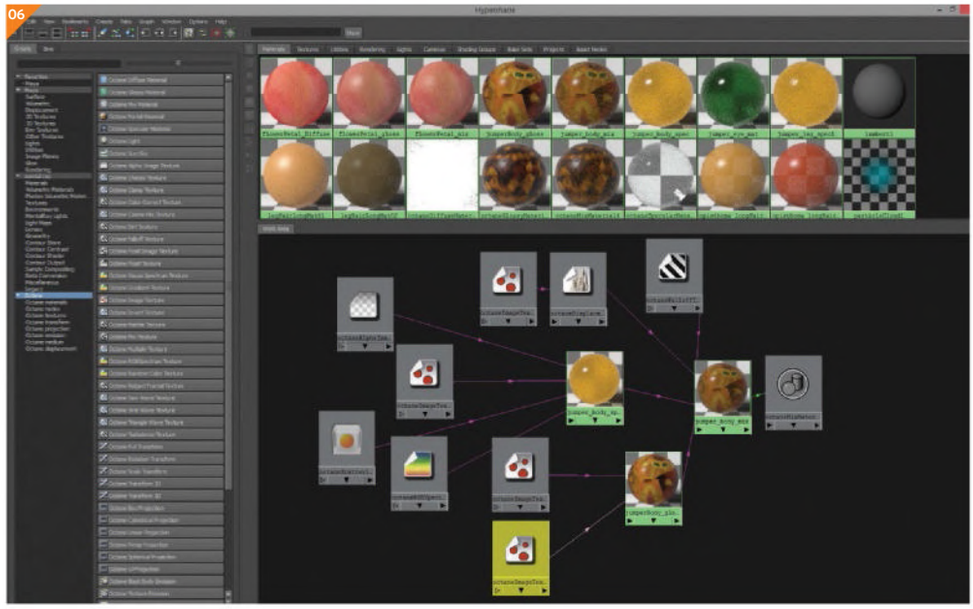
We've used Paint Effects strokes to create the hair for the spider (realistic spiders *must* have hair!). Octane can render Paint Effects without the need to convert the strokes into polygons. You must apply an Octane material to the Transform node of the stroke. Select it and in Attribute Editor expand Octane Hair material. Using the middle mouse button, drag Octane Material from Hypershade into the field to make the connection. If you have hundreds of strokes use an MEL script to make the process faster and easier – we created a simple script for this purpose.

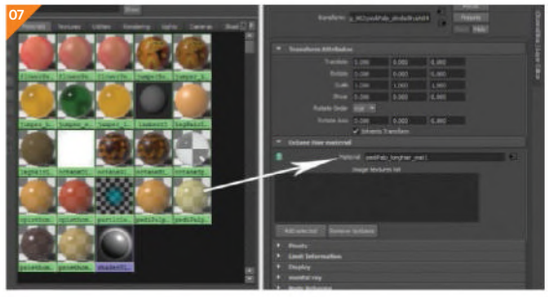
08 Tune render using IPR

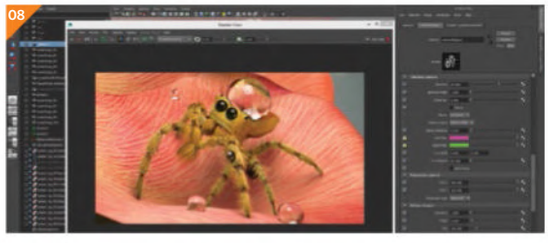
When the materials have been applied and the camera is positioned, you can tune the look of the render using IPR. Select the camera and in the Octane imager section of the Attribute Editor use the Exposure and FStop settings to adjust exposure, allow time for the update in IPR to refresh as the image will get brighter as it renders. To simulate the colour response of real cameras, choose from the list in the Response menu. To add depth-of-field blurring raise the Aperture value and adjust the Focal Depth to set the area of focus (or activate Auto Focus).

09 Set options for final image

For the final render, switch the render kernel and increase the Max samples. The Direct Lighting kernel, which you have been using thus far, is great for previews but to get photorealistic results use either Path Trace or PMC. Path Trace is the best but takes the longest time to render. PMC is best at handling caustics. The kernels have settings to determine how light reacts with scene objects. Max samples can be set to a figure between 2,000 and 4,000, higher values will mean longer render times. The default setting of 16,000 should be lowered. Save your scene and batch render to create the final image.

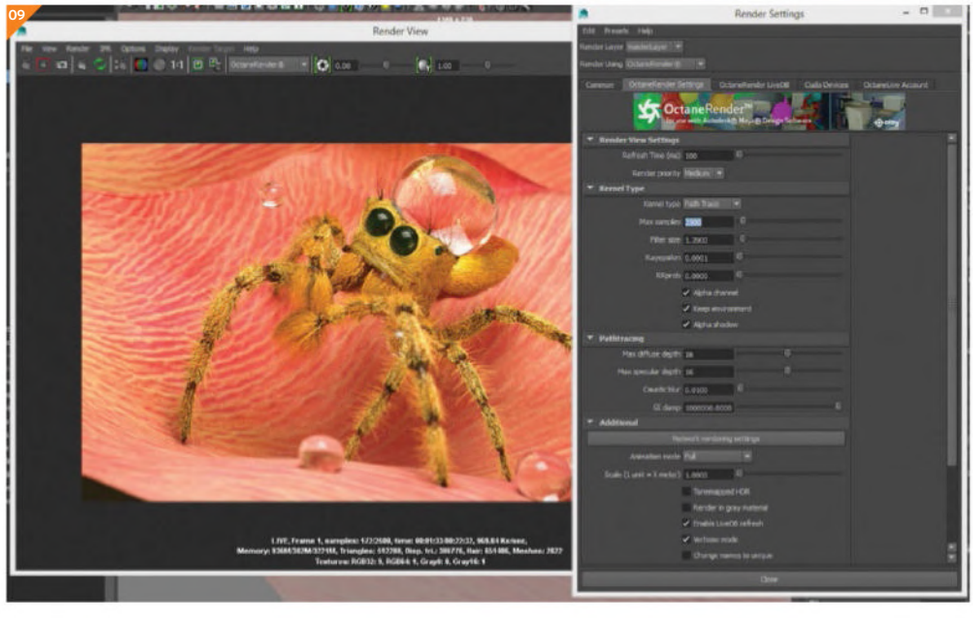






Set the Info Channel

If you would like to render an Ambient Occlusion, Z-Depth, Material ID, Normal, or other passes to use for compositing then make sure that you set the Kernel type in the Render Settings to Info Channel. In the Info Channel type menu select the type of pass you want to render. Specific settings for the channel are found just below the menu. For example if you want to create an Ambient Occlusion pass, set the Info Channel Type to Ambient Occlusion and then adjust to set the maximum AO distance. You can monitor the results in the Render View window using IPR.

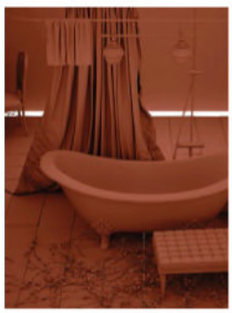


THIAGO LIMA
Pin Up Sex, 2014

Software
Marvelous Designer, 3ds Max, V-Ray, Photoshop

Learn how to
- Integrate photos into a 3D environment
- Correct light in the cutout
- Apply shadows to better integrate the photo
- Cut out a model from a photo
- Light a scene photorealistically
- Work in post-production with Render Elements
- Create extra channels with RenderMask
- Colour correct the scene
- Set up materials
- Set up models and displacement

Concept
This Pin Up Sex project is part of a collection called The Modern Pinups. I wanted to show some pin-up style girls in unexpected situations to push my skills to the limits.



YOUR FREE DOWNLOADS
from **filesilo.co.uk/bks-798**
- Final layered image
- Scene files
- Tutorial screenshots

Blend photos and 3D realistically

Seamlessly blend photographs of people into a 3D environment for a new take on creating unique 3D concepts

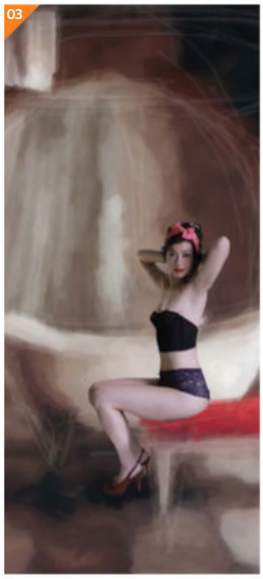
In this tutorial we will show you a workflow that goes through the modelling and postproduction processes of this scene, with creation done in 3ds Max and post-production in Photoshop. Here we'll explain how to model curtains and how to work with high-definition displacements and get a fast and realistic result at the same time. After that we will show you how to set up lighting based on the photo and some materials of the scene. You will also learn how to set up the render settings for a high-quality image with low rendering times. Then we will show you how to render separate selection masks with the plugin RenderMask to be used later in post-production.

Furthermore, we will teach you techniques for a perfect integration between a photo and a 3D rendered image, including cutting out and treating the photo without losing any detail, and correcting and balancing the lighting, vibrance, speculars and shadows. The process and workflow used in Pin Up Sex will help you in your own daily projects.

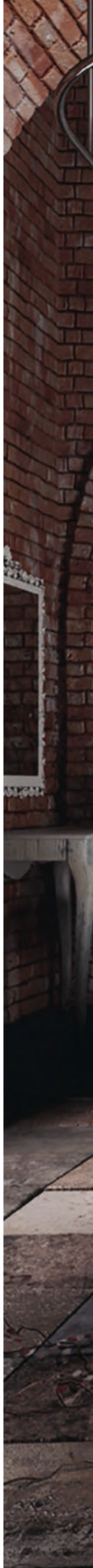
01 Choose your path Because this work is part of a bigger collection of images, some things had to be taken into account such as a recognisable identity between all the different images. All of the images in the collection contain a subliminal message: something obvious that is sometimes disguised among other elements of the scene. This is the third image collection in the series and the idea is that every new image needs to have some kind of challenge to be overcome. The greatest challenge in Pin Up Sex was to compose one person photorealistically and naturally in a 3D render that also has light sources coming from different directions.

References: use without moderation
Remember that for everything that is created in these scenes (in terms of modelling, lighting, materials, textures and even the mood for the postproduction), several references have been used. It is a common mistake to forget about them because our brain normally has a distorted memory of reality, therefore it's not always reliable to trust our memories. Just take care when you use a whole piece of work as reference, you should ask for usage permission or credits, and the same for if you wanna use some cut-out people in your scene.

02 The shot This is one of the most important parts of this project. The photo of the model, Daniela Donato, was taken by herself and has no professional production – a lot of people don't think they can do high-quality projects without professional equipment. In the picture we have three main sources of light, one behind the camera, one over the table and one on the right side. This setup was intentional to practice a more complicated light setup for the 3D part.

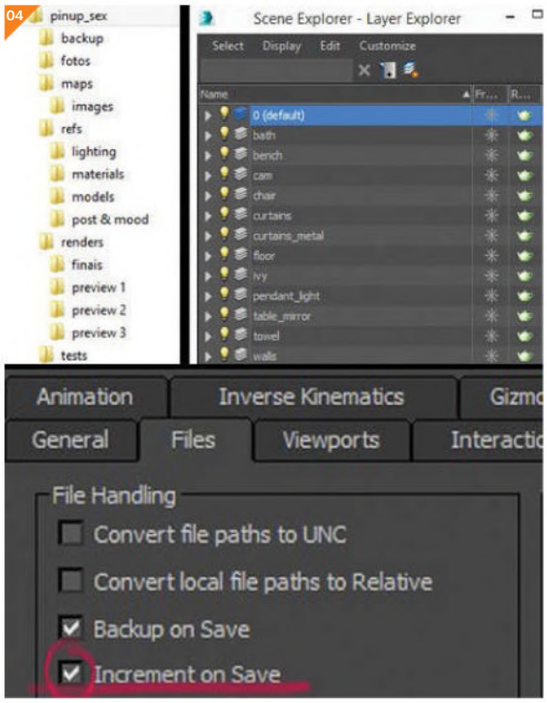


03 Do some sketches When you start to think about practical production, the first step is to do some sketches using the photo as reference, to inspire yourself and freely create your layout independent of premade assets. Also it's nice to think about what kind of concept, composition, storytelling and emotion you can bring to the viewer. Ask questions to yourself like: is she alone in this place? What is her mood now? Why is she there? Every new answer can guide you to a different composition of the same main concept.

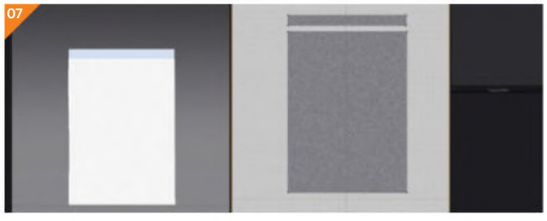


"It's nice to think about what kind of concept, composition, storytelling and emotion you can bring to the viewer"

04 Organise your files Before starting any project it's advisable to have a good organisation of your folders, especially regarding references and maps. It is also advisable to use layers to find and group objects in your scene as well as the use of the 'Increment on Save' option located in Customize>Preferences. This creates a new file every time your file is saved, leaving a backup in case of some mistake, a crash which corrupts your file or if you decide to go back to a previous stage of the project.



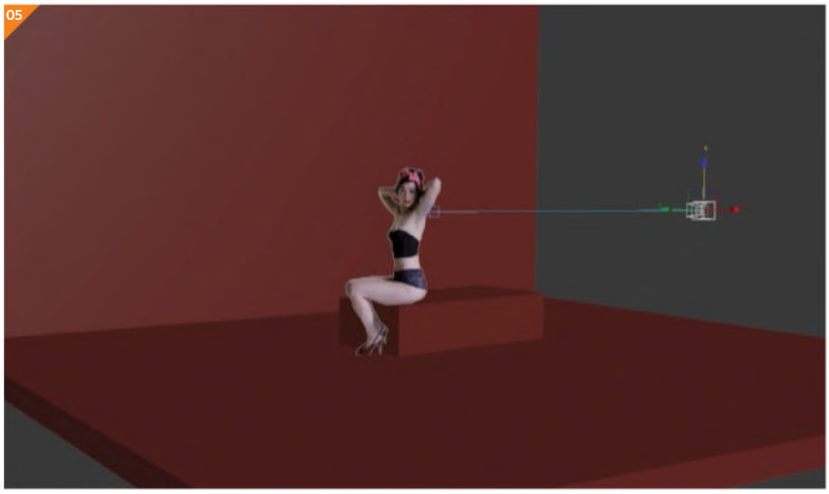
07 Start the curtains Inside Marvelous Designer 4, delete the default Avatar by going to Avatar> Clear all avatars. In the 2D viewport create two similarly width rectangles, a small one on top to serve as support and a larger one below which will be the curtain. Right-click on 2D View>Show Mesh to see the subdivision of the mesh, use the tool Sewing Segment and click the bottom edge of the small rectangle and the upper edge of the large rectangle to attach them with the seam. Right-click on the rectangle small in the 3D viewport, and then click Freeze. Press Space to simulate.

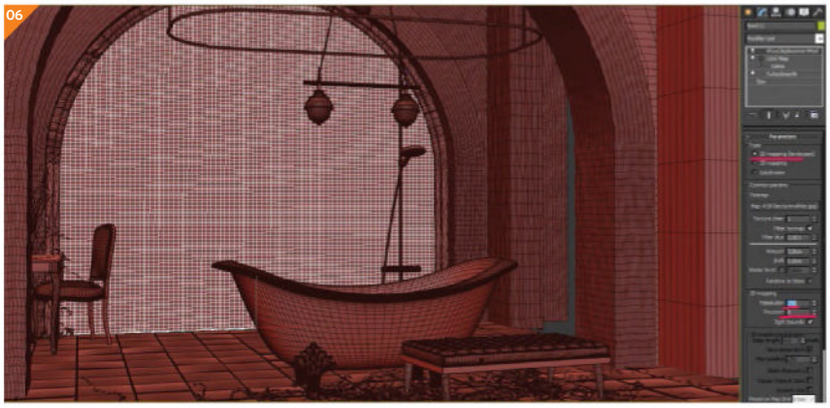


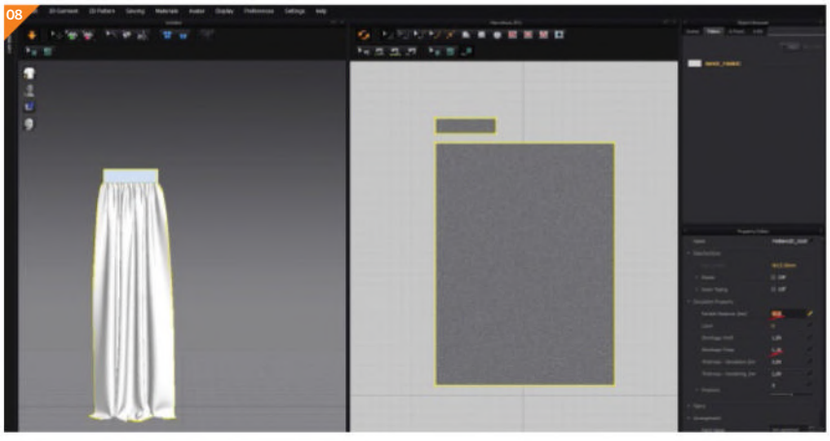
08 Finish the curtains Decreasing the width of the smallest rectangle, which is in Freeze mode, will wrinkle the curtain. Select both rectangles in 2D View, and decrease the value of Particle Distance to a subdivision you like. If your curtain is not touching the ground you can slightly increase the value of Shrinkage Warp to 1.10. Click and drag the curtain in the 3D View to position it as you like by holding W to create a pin to keep it there. Now export as an OBJ using the unit or scale that you are using in 3ds Max.

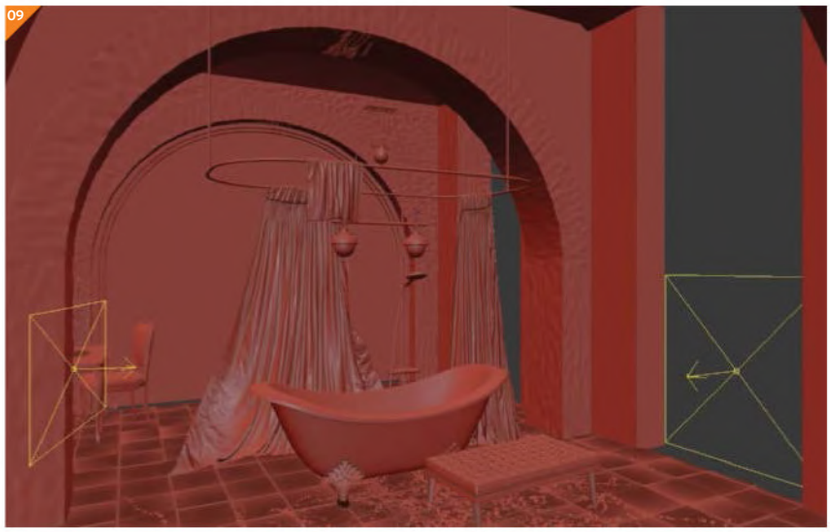
05 Set up the camera As we are going to create the whole scene based on the shot taken of the model we need to first set up the camera based on the camera of the original photo. You just need to know three simple things: the camera lens of the photo, the camera distance from the model and the camera height from the floor. So, you can imagine how hard it is when you're not the photographer because you need to guess everything. Depending on the scene you can also use Camera Match in 3ds Max to easilly create your camera.

06 Modelling and displacement In the modelling process you can keep it simple.. Most parts of the objects had cubes as a base mesh and the floor had to be done separately as single tiles for easily varying the mapping and textures with the MultiTexture plugin later. You may have noticed that due to the displacement the subdivisions of the brick walls are very dense. This is because when you increase the subdivision of objects with displacement you basically ease the processing at the time of rendering so that you can use lower settings while maintaining quality and optimising rendering times.

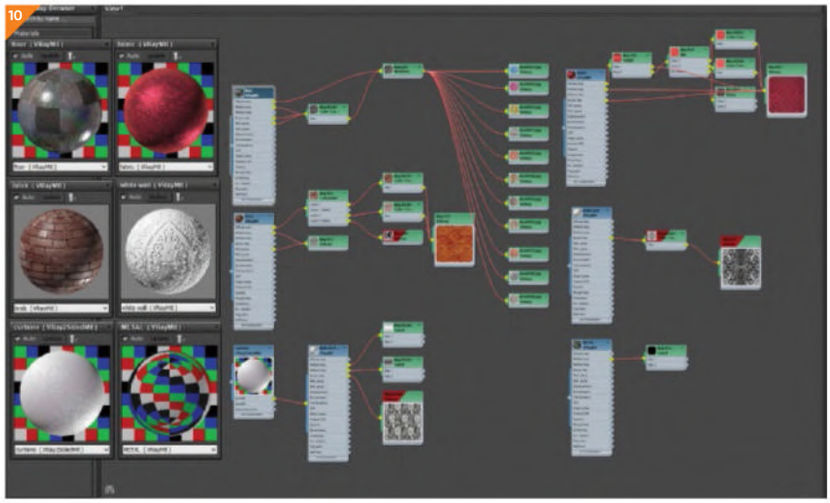








09 Light the scene As we had control over the photograph taken by the model in this project, the lighting process for the 3D became much easier because we knew the exact amount and positioning of the lights in the photo stage. The parameters of the VRayLights are almost default, just modify the left light leaving it weaker than the light from the right. The light coming behind the camera will be created just from the light bouncing over the scene. The values of the light intensities are of the least concern since this intensity will vary widely due to the exposure of the camera and scene modelling.



10 Materials and maps Always keep the materials of your scenes as simple as possible; for almost every scene material that was used, only a map of Diffuse, Reflection and Glossiness were used with it. It is not always necessary to use a Bump map; the Reflection and Glossiness can provide a Bump-like effect, which helps to accelerate the speed of your render. You can see in the picture for this step that the MultiTexture map is applied to the floor material and also the simplicity of other materials. Try to keep eight subdivisions in all materials, increasing only if some noise occurs.

A drop of theory

Sometimes it may seem difficult to create a convincing and realistic material. In bases you need to look at just a few things, so, gather all the reference materials you can for what you want to create, in various types of lights and camera positions, and see how it behaves with parallel and perpendicular reflections – this reflective index will give you the intensity of Fresnel in your V-Ray materials. After that, use Reflection and Glossy maps to create a variation of areas that are more or less glossy or reflective. Control the percentage of influence on each texture in the Maps tab.

Showcase

Thiago Lima

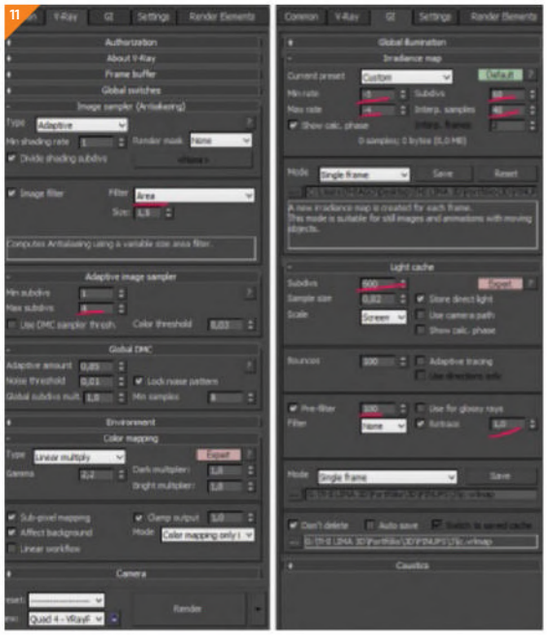
I am a graphic designer who started with 3D as a hobby seven years ago. After that I started to merge CG with my 2D works. Currently I am a generalist CG artist focused on workshops, lectures, online courses, consulting in CG field and photorealistic 3D freelance jobs for architecture and VFX.

The Last Pinup **3ds Max, Corona, Photoshop (2014)**
This scene depicts a moment from an alien invasion. This work was done to test the limits of Corona.

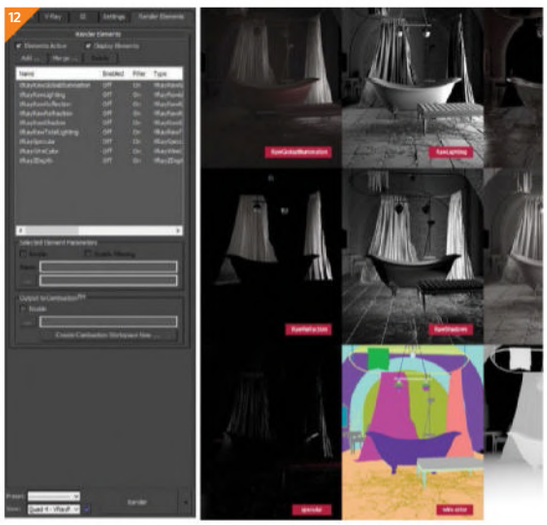
Pin up, Le Dernier Cafe **3ds Max, OctaneRender, Photoshop (2014)**
This scene depicts an apocalyptic scenario where a giant tsunami carryiynigng a transatlantic ship is invading Paris. This work was done to push the limits of OctaneRender.

Butterfly **Marvelous Designer, 3ds Max, V-Ray, Photoshop (2013)**
In this project I've created a home in a new way – it looks like half a butterfly and the other half is mirrored in the reflection.

11 Render settings We must understand render settings as a tool in order to optimise render time and fix errors, and not as something that will increase the realism of our scene – a render setting's high values do not result in a greater realism. So we can use very low settings remembering that the better lit your scene is, the fewer values you will need to increase in the render settings. A great tip with the antialias filter is you can use the Area filter with a size greater than 2.0 to bring softer edges with a real photography-like edge sharpness, depending on render output size.



12 Render Elements Here you can see how Render Elements is used in our post-production. The VrayRawLighting for example is one of the most important because we have a Direct Light channel without the influence of global illumination – it can increase the amount of direct light or even the overall contrast in a very efficient manner. You can use the plugin RenderMask to create an extra Alpha channel which will save you from doing a lot of work when editing objects separately. Don't forget to set the Min and Max Distance of the Z-Depth channel based on your scene depth.



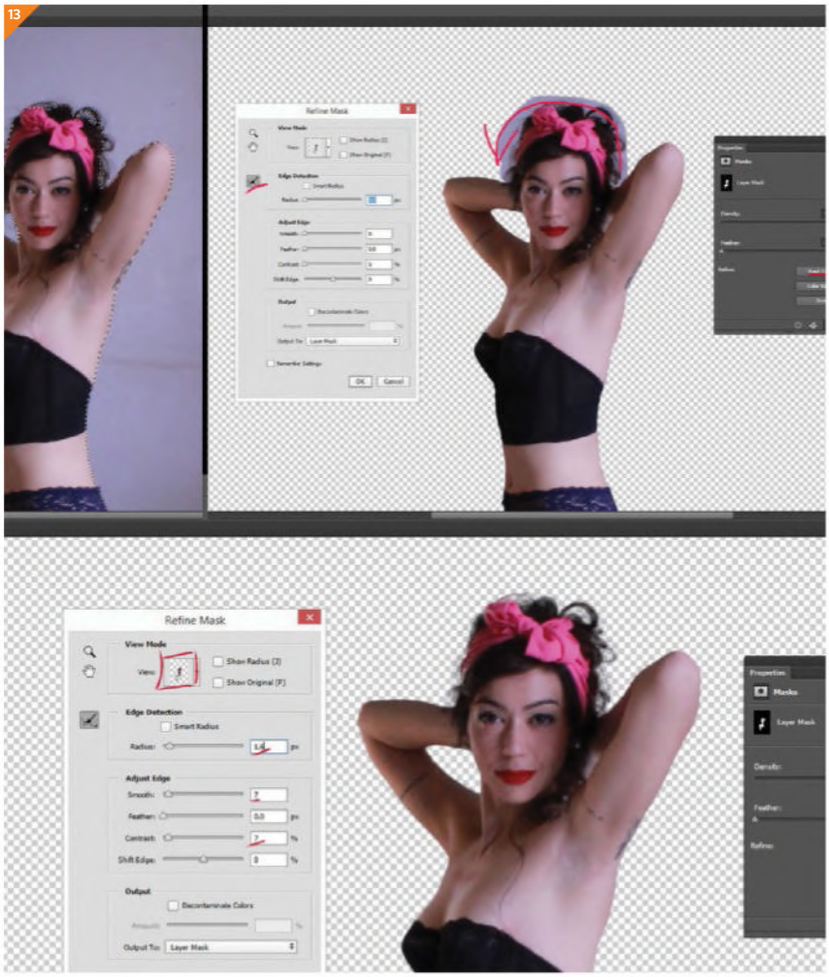
> Global illumination can increase the amount of direct light or even the overall contrast in a very efficient manner

13 Cut out the photo The cutting out of the model is not so easy, especially in the case of the small hair strands. A good way to start is using the Quick Selection tool to make a rough selection of the model, click Add Vector Mask in the Layers menu, double-click the mask created and then click on Mask Edge. Now brush over the scene with the Refine Radius tool to add the tiny hair strand details to the selection. You can play as you wish with the other options to better define your selection, the Shift Edge is one of the most important options as it controls the selection expansion.

14 Use Render Elements You can start working with the light and contrast using the VrayRawLighting element with the blend mode in Soft Light, and use RenderMask to change the desired areas. Change the Opacity to control the incidence of the effect. The amount of time that you choose to work on each element is purely up to you. Furthermore, balance the levels of white and black, and saturation, then think about the importance of each element in the scene with regards an eye's natural path. For example, in this scene, a layer was created to desaturate the bench so that it balances with other scene elements.

15 Compositing the girl To finish, add the Z-Depth Channel as an inverted overlay, it will make the background brighter than the foreground and direct the viewer's eye, and adjust brightness or contrast again. Place the cut-out model in the scene and adjust the colours, brightness, saturation using the Levels, Selective Color and Hue/Saturation. For the shadows select the model, paint them black in a new layer, reposition them and then add a Gaussian Blur to smooth the edges. Play with Multiply, Soft Light or Normal blend modes to see which fits better. Make the final adjustments with the model's colours and contrast.

13

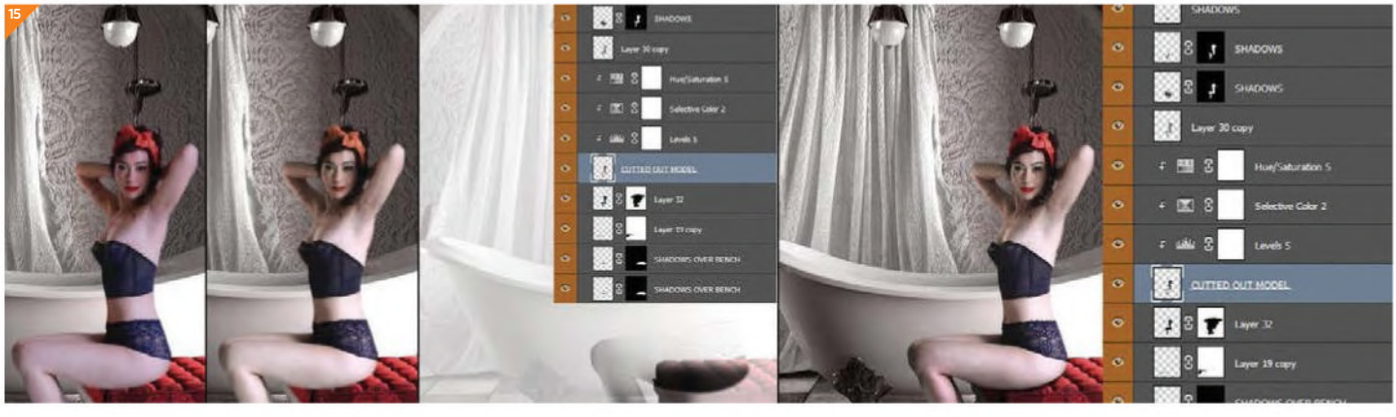


14

Post-processing the model

Use the cutout file to make any necessary adjustments such as treating the skin, or removing blemishes, noise, colours. In this case it was doing an interesting brief treatment of the clothing colours and also the addition of higher sharpness in the image – for the latter you can use a very simple technique. Copy the layer of the woman, then apply the Filter>Other>High Pass. Change some of the values, keeping only the edges pronounced by using layer blends in the Overlay mode. You can control the intensity of the effect by changing the opacity.

15

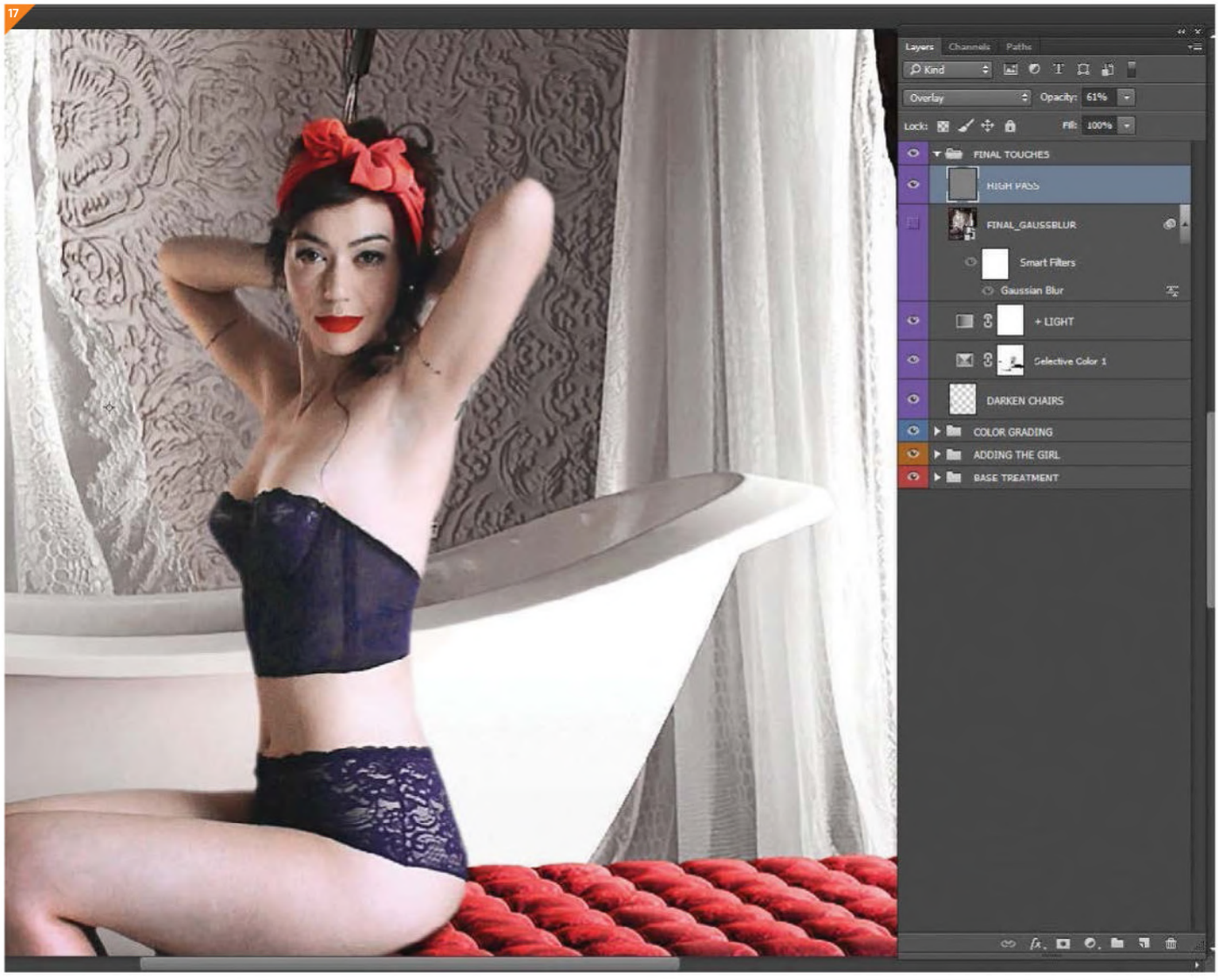


16 Colour grading This is the best part of the job when we can play with the colours and the emotions that the scene can provide us with. For the contrast you can use the Gradient Map Adjustment layer in Soft Light blend mode and tweak inside of those two. For colours you can use Curves, Color Balance and Selective Color, and work in each colour channel individually using shadows, midtones or highlights. Another thing to worry about is the level of sharpness of the model and the 3D scene, as both must be perfectly balanced so that it looks like a natural, seamless image. Also, use levels to tweak the whole composition equally.

17 Final compositing Well, is time to fix everything that you had previously left behind, and maybe return at some stage to anything that you may have exaggerated. In this case, it was necessary to add a little more darkness in the background chair so that it draws less attention in the scene. After that use Selective Color to bring the model's skin colour back. Finally you can copy all layers and merge all the copied ones – right-click on it and then use Convert to Smart Object so that any effect applied to this layer will be editable. You can use a Gaussian Blur and then increase the sharpness by using the HighPass filter.

Conclusion

It's always a good idea to organise yourself well at all stages, from modelling all the way through to completion. Try to work in sections, for example when creating your materials, you can make a scene with your desired lighting and a white material, and then you can create the materials one by one – just skip to the next material after the first one seems perfect for your needs. Never forget about the references, as they will save you time. Finally, do not be afraid to make mistakes: to change, lift off rules, numbers, parameters and so on. Photoshop can be awesome, so don't waste so much time on trying to get perfect raw renders.



> Everything from subsurface scattering, to caustics, to colour grading is all done in camera within a single pass

Render still life with RenderMan

Get a preview of the key features in the upcoming release of RenderMan version 20 and understand common workflows

This tutorial will focus on lighting and look development in RenderMan for Maya. The goal is to introduce RenderMan's new RIS technology and show how the simplified workflows can deliver photorealistic results quickly. The objects that can be seen in this still life were chosen specifically to demonstrate a broad range of effects, concepts and workflows.

This tutorial also serves as a preview for many key features available following the release of RenderMan 20. Pixar launched the new version in mid 2015, and this is a strong overview of a few of the newest tools. While RenderMan is quite capable of producing custom secondary passes for compositing, in this tutorial everything is rendered 'in camera' during one pass. There are no prepasses such as shadow maps, point clouds or whatnot – everything from subsurface scattering, to caustics, to colour grading is all done in camera within a single pass. There is no use of NUKE or Photoshop, and the only postprocess is the amazing new Denoiser (which we will talk about in more detail in Step 15).

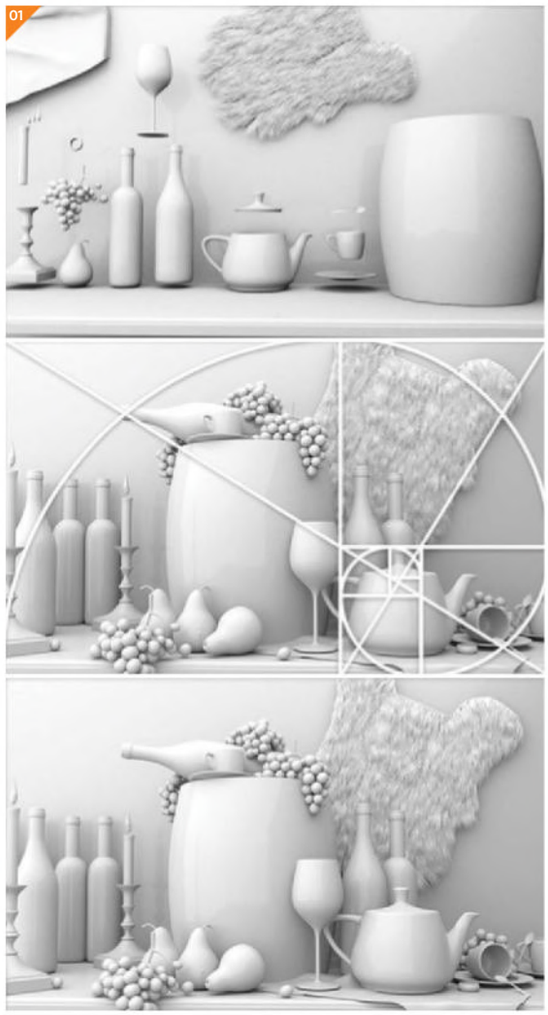
Following the release of RenderMan 20, you can try it out for yourself. If you're not already a RenderMan user, you can download the Free Non-Commercial RenderMan to follow the tutorial.

DYLAN SISSON
Arrangement with Utah Teapot No1, 2015

Software
Maya, RenderMan

Learn how to
- Do interactive rendering
- Light a scene
- Develop a look
- Apply camera effects
- Use the Denoiser
- Render fur
- Use emissive geometry

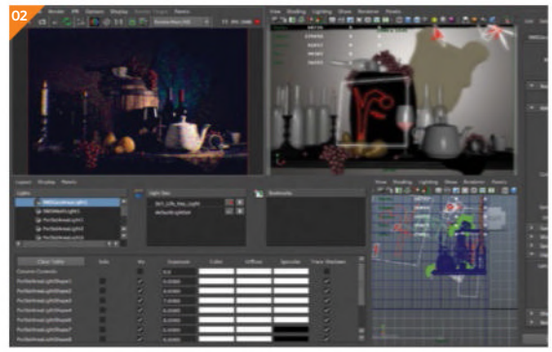
Concept
The hero of this still life is the original Melitta teapot that Martin Newell purchased in 1974 in Salt Lake City, Utah, which he used as reference to create the popular CGI model, the Utah Teapot.



02 Interactive rendering The majority of this project was created during interactive rendering sessions to leverage the acceleration that RIS technology provides for both lighting and look development. Interactive rendering makes lighting setup fast and natural. The special RenderMan Lighting Panel makes working with lights efficient – all you have to do is just edit the intensity of groups of lights all at once and bookmark your favourite light settings as desired. From the UI, you have the option of starting an IPR session in Maya's Render View or using Pixar's own image tool 'it'. Interactive rendering can be a real time-saver.

03 Geometry It's always a good idea to check out the geometry in your scene at the beginning of a project and make sure that there's nothing too abnormal lurking about, and an upcoming feature that will help us with this in RenderMan 20 is a new Visualizer Integrator which enables scene geometry to be viewed the same way that RenderMan 'sees' it. You can visualise things like wireframes, normals and so on. Because the Visualizer is extremely lightweight it can be used to manoeuvre through highly complex scenes during an interactive render. So first, you just need to start up an interactive render and then go on to select the Visualizer Integrator from the Sampling tab in the globals and look at the scene. For our project the wireframes check out, but if there were issues with the geometry or parameterisation, that would also be apparent to us here.

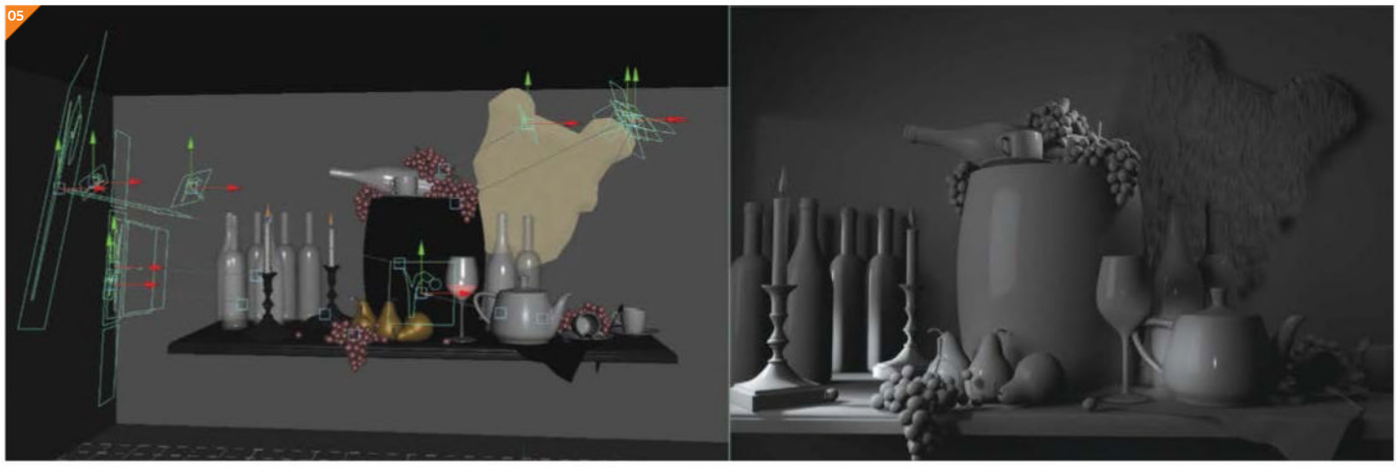
01 Compose the scene The main subject of the still life is the teapot, which was modelled using photo references from the internet. Other models were then added to the scene to show specific features and effects: a rug to show hair rendering, fruit to show subsurface scattering, a candle to show emissive volumes and so on. This became quite cluttered, so to properly direct the eye it was important to create a compelling composition. For this particular image, the golden ratio was used to guide the placement of the objects. An image of the golden ratio was attached to the camera plane in Maya and used to quickly lay out the objects.



Subdivision Surfaces
Pixar did the original research and development for subdivision surfaces, and recently promoted this standard for subdivision surfaces with the Open Subdiv project. Open Subdiv is a set of open source libraries adopted in Maya and across the industry – the subdivision surface you see in Maya will be the same as RenderMan's. You may choose to work with Maya's subdivision surface primitives or simply work with a polygonal mesh and render it as a subdivision surface by attaching special attributes to the geometry (the method we used here). RenderMan translates creases on poly meshes, so complex topologies like the candlestick can be modelled. Tip: for easy setup, try smoothing any poly mesh (by pressing 3 in Maya) and RenderMan will treat it as a subdivision surface – no extra attributes required.

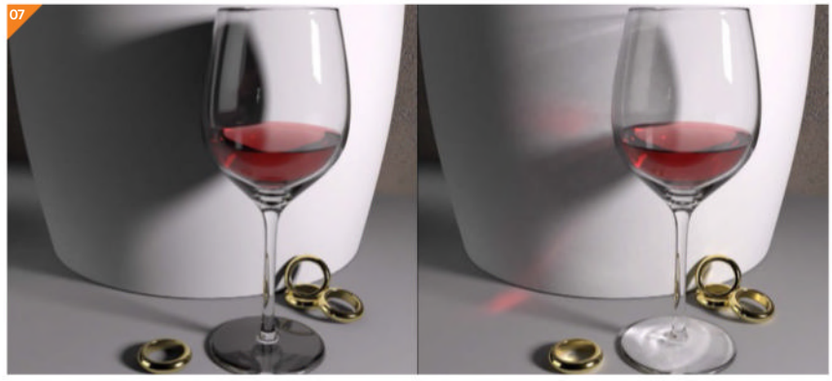
YOUR FREE DOWNLOADS
from **filesilo.co.uk/bks-798**
- Maya Scene File
- Tutorial screenshots

04 Make the Key Light First, create a Key Light to direct the viewer's eye. For maximum control and efficiency RenderMan has its own specialised lights, and we'll create one from the RenderMan menu (PxrStdAreaLight) as the Key Light. Start up an interactive session (RenderMan>IPR Render) and select the light. To create a dramatic effect, position the light to the right of the table and focus it on the teapot like the image. Now dial in the lighting for the key: increase the exposure, adjust the colour temperature, narrow the light profile to tighten the illumination, and enable 'barn doors', which provides fine control for constraining the light.



05 Create fill lights To emphasise different elements of the composition we can use fill lights to highlight specific objects. With the interactive session running from the previous step, simply create more lights and then position them as you see fit – for example if you want to fill in the teapot, to cast highlights on the grapes or to add rim lighting the candlestick. In order to 'paint' a specific object with light, like the teapot for example, an extremely tight Light Profile can be used (such as a setting of 10) which will enable the light to illuminate the teapot, but not any of the objects that are adjacent to it.

06 Lighting transparent objects Next, make sure that the wine glass renders properly. In order to achieve good results in the most efficient way, make sure you pay attention to the 'Max Specular Depth' setting in the Sampling tab of the globals. For a ray to travel all the way through the wine glass, it must pass through at least two surfaces before exiting. In the image you can see the results from a 'Max Specular Depth' of 2, 4 and 8. If Max Specular Depth is set incorrectly you may see black areas in the glass. Note that 'Max Specular Depth' can be assigned on a per-object basis from the RenderMan Attribute menu.



07 Create caustics For the wine glass, it's desirable for light to refract through the glass, but notice that when the Path Tracer integrator is enabled via the Sampling tab in the globals, the shadows that are cast by the wine glass are solid black. Path tracers simply aren't that good at rendering caustics, so RenderMan provides a VCM integrator which uses bidirectional path tracing to efficiently resolve caustic effects for both transparent and reflective objects. Switch to the VCM integrator and that's all you need to do. In the image you can see caustics are automatically generated for the wine glass and the golden rings.

RIS and Reyes

It is important to be aware that RenderMan essentially has two rendering modes, RIS and Reyes, within the same software. This tutorial uses the new RIS renderer. RIS is highly optimised for rendering global illumination, specifically for ray-tracing scenes with heavy geometry, hair, volumes and irradiance using single-pass workflows. Because of differences in their architectures, it's important to be in RIS mode when following the tutorial. While Reyes adds additional functionality, RIS is completely new technology that has been developed to take full advantage of modern hardware for physically based ray tracing, featuring intuitive tools, simplified workflows and photorealistic results.

Showcase

Dylan Sisson

Dylan Sisson combines 15 years of experience in 3D animation and VFX with a traditional background of painting and illustration. He's created paintings, illustrations and vinyl toys that have been shown in galleries around the world, and his independent animated shorts have won several awards.

Idle Hands **ZBrush, Maya, RenderMan (2015)**
Idle Hands is a limited edition vinyl toy collectable designed and sculpted by Dylan Sisson and produced by Toy Tokyo.

Walking Teapot **Maya, RenderMan (2014)**
The RenderMan Walking Teapot was created by Dylan Sisson. This rendering is a mockup created for the 2014 model design.

My Tie **Corel Painter (2013)**
Part of a 'cocktail recipes gone wrong' series, this art coaster was made digitally and printed on a letterset press.

08 Subsurface scattering for the teapot The main quality of the ceramic Utah Teapot is its translucent surface, which we can replicate with subsurface scattering. In this still life, subsurface scattering was used as the base material for the teapot, teacups, grapes, pears, candles and wooden barrel. To simulate any of these surfaces, just use RenderMan's Layered Material System to create an LMSubsurface material. Three controls are provided for subsurface scattering (near, middle and far). With the addition of two specular lobes, with their own independent bumps, the LMSubsurface is capable of all sorts of looks. With interactive rendering, dialling in the right look is fast and intuitive.

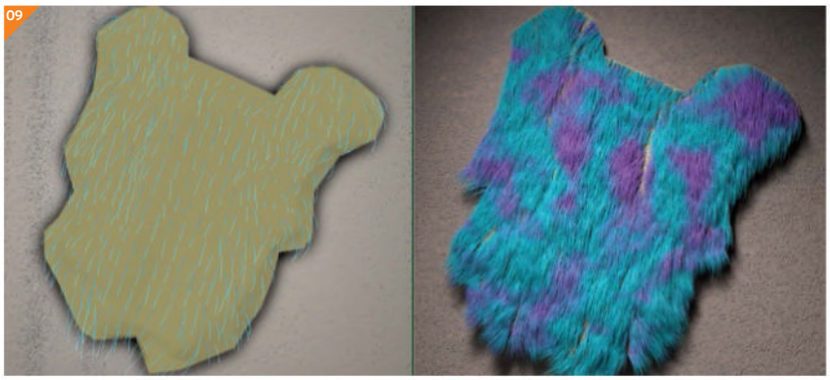

08

09 Wall rug – render Maya Fur First of all this is faux fur – no cartoon characters were harmed in the creation of this fur. RenderMan for Maya automatically renders Maya Fur and Hair, but what makes this fur special, is that it is rendered with a feature from the upcoming release – the new Marschner Hair shader created for Pixar production. That's correct, this is the first shader to ship from Pixar that was developed for an upcoming Pixar feature film. Marschner Hair delivers movie quality results and setting it up is simple – just attach the Custom Shader attribute to any Maya Fur description and add Marschner Hair.

10 Wine barrel – layer materials Here we'll use a layered material to create our wooden barrel bound together with metal bands. The geometry itself is simple, just a lightweight cylinder that flares out in the middle. First, a displacement shader is used to create the additional features using a texture map. Next, the base layer for the wood is created using an LMSubsurface node and it's given just the right amount of subsurface scattering and specular highlights to simulate wood. Finally, another layer is added on top of the wooden material to simulate metal, and it's then masked so that the bands appear metallic.

11 Candle flame – emissive volumes The candle flame was created using nested volumes and emissive geometry. To create a candle flame, first create an elongated sphere and attach a Volume Shader (pxrVolume), then on the Shading Group add a custom 'Area Light Shader' attribute and connect a Mesh Light. Now the elongated sphere will behave as any other light source, even though it is a volume with distorted topology. Next, place a slightly larger volume sphere around the first sphere to create the orange glow. Finally, add a sphere near the bottom of the flame and attach a blue volume shader. It's a quick method of building a flame.

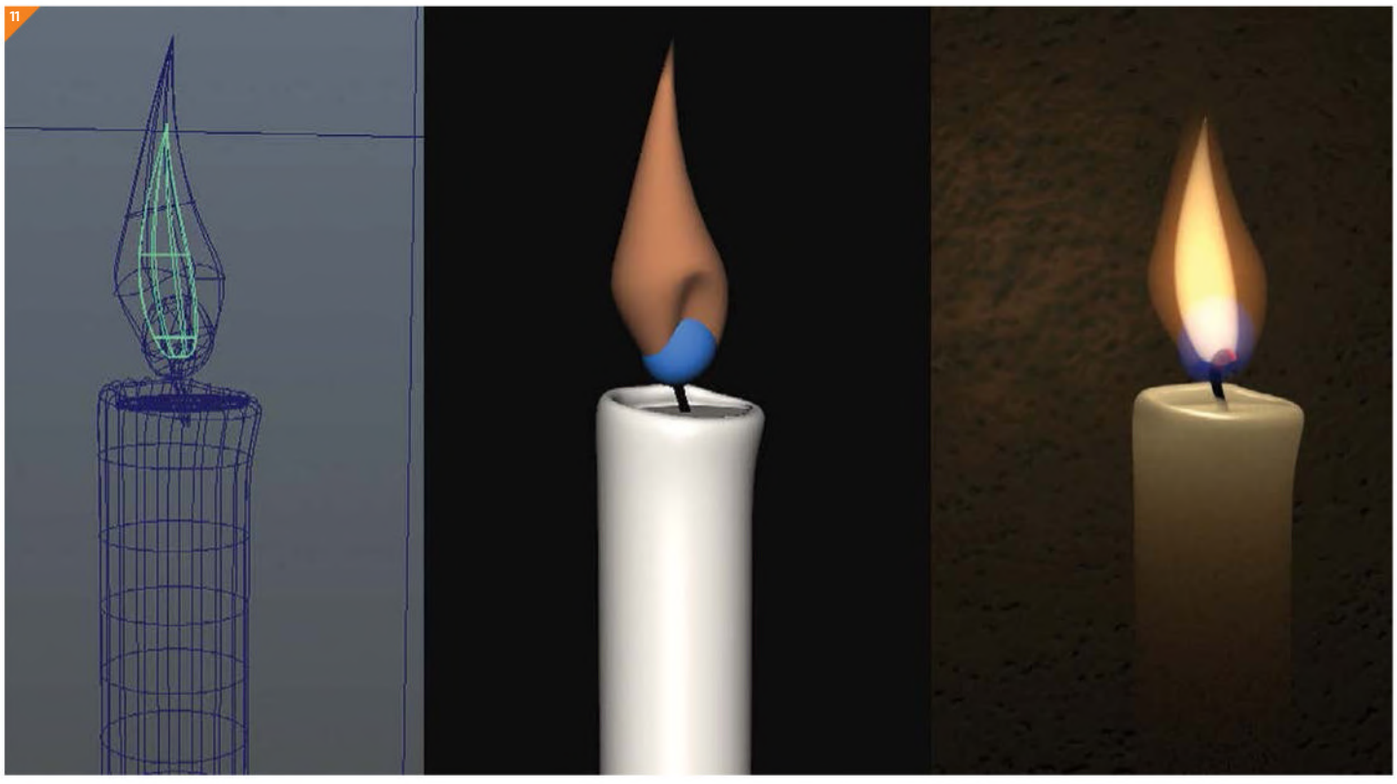
09

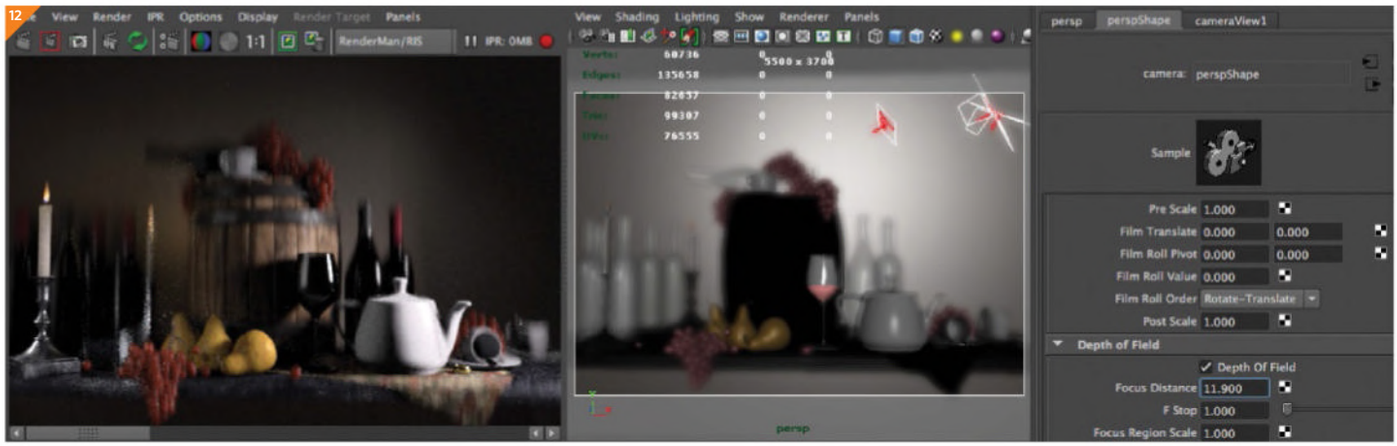


10

> This is the first shader to ship from Pixar that was developed for an upcoming Pixar feature film

11



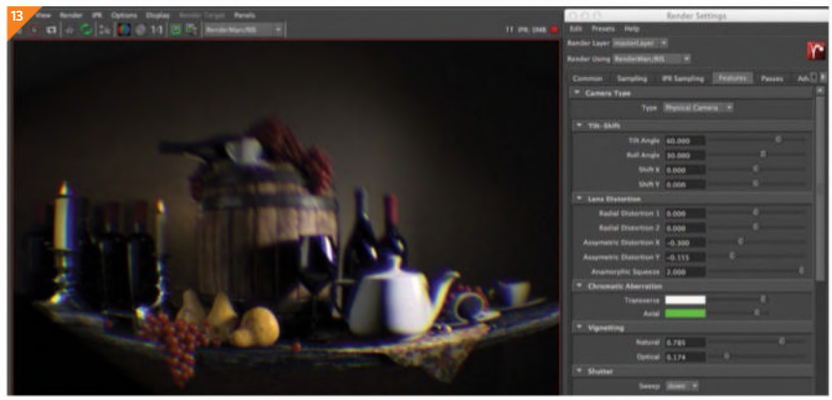


12 Get some focus With the scene nearly ready to render, it's time to add depth of field. Just enable depth of field in the Maya camera and RenderMan for Maya will automatically render it. Special RenderMan Aperture controls can also be added to any camera via the Attribute Editor, and these provide support for features like faceted bokeh and oval-shaped defocus. Tip: when dialling in depth of field, it can be useful to use the interactive renderer and for heavy scenes it can be fastest to use the Visualizer Integrator mentioned above to set the focus. For the still life it was important to keep the teapot in focus and slightly blur the background.

13 Create a physical camera In the real world, cameras are imperfect and can cause artefacts like vignetting. However, these imperfections can also give an image character. In the upcoming release, RenderMan now has a true Physical Camera which supports vignetting, chromatic aberration, lens distortion and tilt shift photography. For this still life the physical camera was an essential tool that was used to give the final render a look that is closer to a real photo. To set up the effect, simply enable the Physical Camera in the Features tab of the globals and adjust the settings as you see fit.

14 Add colour grading RenderMan has a class of shader called 'imager' which can be run at the end of a render, across the entire image. A common use for imager shaders is colour grading, for those times when it is desirable to render everything in-camera. This somewhat advanced step will involve loading an imager shader (slo) into RenderMan for Maya using a 'Frame RIB Box'. In the case of the still life the imager shader was created in Pixar's Slim, and this was then used to push the shadows toward blue and the highlights toward orange. With the imager set up correctly, we're now ready to render.

15 Render with the Denoiser Finally before pressing render, enable Denoise in the Sampling tab in the globals. The Denoiser was developed to address the issue of noise in physically based rendering (where images can take a long time to converge to a completely noise free image). Now, renderings can use fewer samples for faster renders, and when the rendering is complete the Denoiser is run on the image to remove any artefacts. RenderMan for Maya writes out additional information (AOVs) into a large multichannel image (EXR) for the Denoiser, which enables the Denoiser to only remove the noise. The Denoiser might just be the 'Make Pretty' button we've all been waiting for.



More questions?

This tutorial is just a start. A lot of ground has been covered to introduce some of the features in RenderMan's RIS technology and yet we've only scratched the surface. There simply wasn't enough space to thoroughly show each and every workflow, and the focus was placed instead on conveying a breadth of techniques. For further exploration, please refer to the scene files of the still life on FileSilo where you can closely examine the setups shown in this tutorial. RenderMan is also free for non-commercial use, so it's easy to begin exploring on your own. For more RenderMan resources please see **renderman.how**.

YOUR FREE RESOURCES

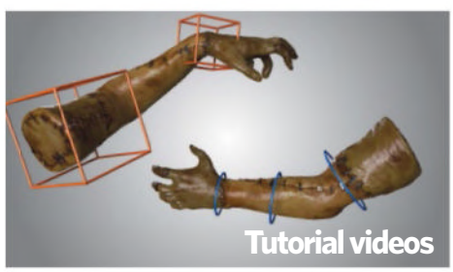
Log in to **filesilo.co.uk/BKS-798** and download your 3D resources **NOW!**

EVERYTHING YOU NEED TO FOLLOW ALONG WITH THE BOOKAZINE AND CREATE GREAT 3D ART

11 HOURS OF DIGITAL-TUTORS VIDEO TUTORIAL

Tutorial scene files

Premium CGAxis models



Tutorial videos

THIS EDITION'S COMBINED RESOURCE SIZE: 23GB

YOUR BONUS RESOURCES

ON FILESILO WITH THIS BOOKAZINE, FREE AND EXCLUSIVE FOR THE 3D ART & DESIGN ANNUAL READERS, YOU'LL FIND OVER 23GB OF INCREDIBLE RESOURCES, INCLUDING...

- All the tutorial files necessary to complete the projects in this bookazine
- Premium CGAxis models
- Over 11 hours of quality video tuition from the experts at Digital-Tutors
- A wealth of inspirational images from amazing professionals
- Please note the content on FileSilo.co.uk is subject to change

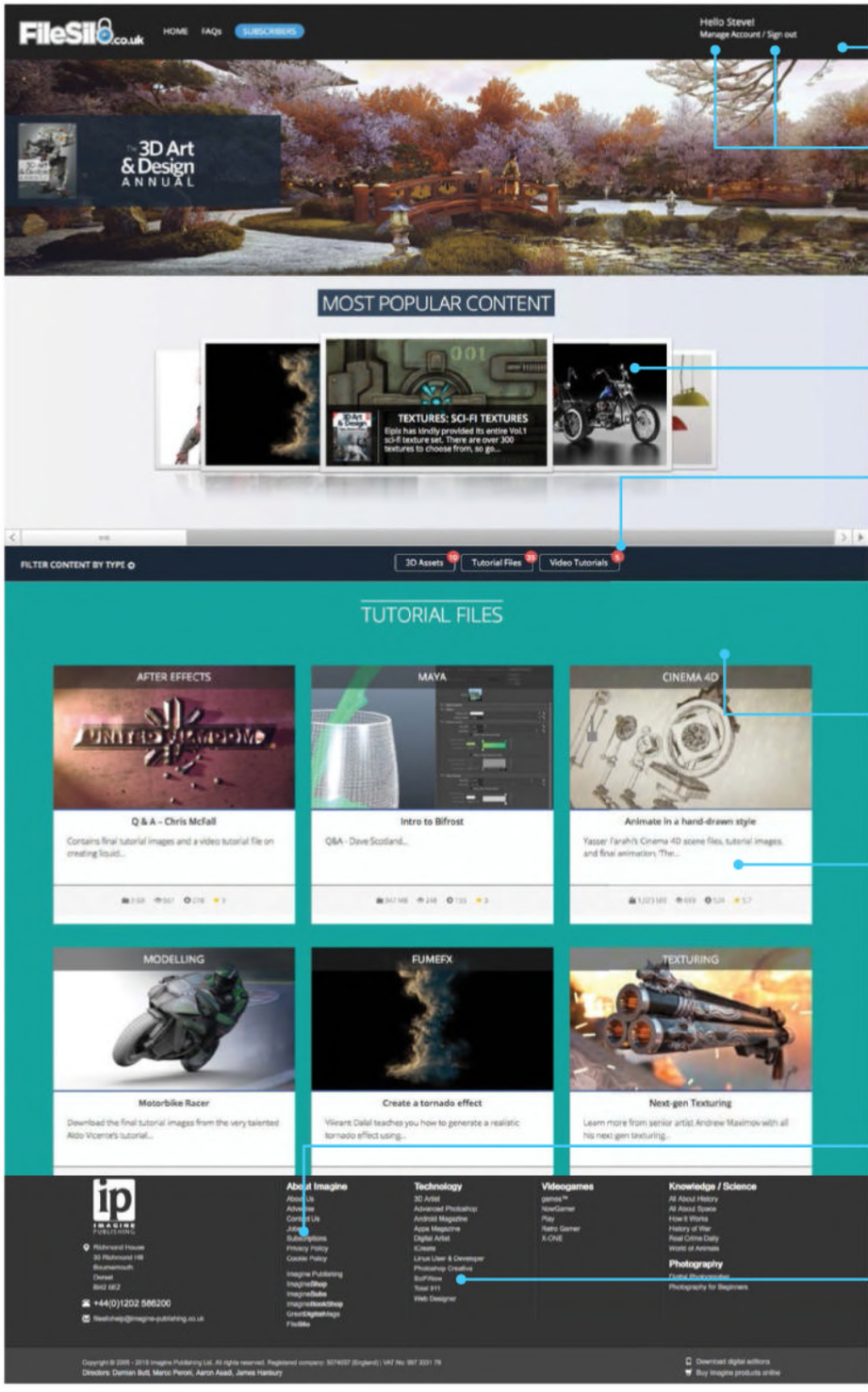
FileSilo

Go to: filesilo.co.uk/BKS-798

FILESILO – THE HOME OF PRO RESOURCES

Discover your free online assets

- A rapidly growing library
- Updated continually with cool resources
- Lets you keep your downloads organised
- Browse and access your content from anywhere
- No more torn disc pages to ruin your magazines
- No more broken discs
- Print subscribers get all the content
- Digital magazine owners get all the content too!
- Each issue's content is free with your magazine
- Secure online access to your free resources



This is the new FileSilo site that replaces your disc. You'll find it by visiting the link on the following page.

The first time you use FileSilo, you'll need to register. After that, you can use your email address and password to log in.

The most popular downloads are shown in the carousel here, so check out what your fellow readers are enjoying.

If you're looking for a particular type of content, like software or video tutorials, use the filters here to refine your search.

Whether it's tutorial files or video workshops, categories make it east to identify the content you're looking for.

See key details for each resource including number of views and downloads, and the community rating.

Find out more about our online stores, and useful FAQs, such as our cookie and privacy policies and contact details.

Discover our fantastic sister magazines and the wealth of content and information that they provide.

HOW TO USE FileSilo

EVERYTHING YOU NEED TO KNOW ABOUT ACCESSING YOUR NEW DIGITAL REPOSITORY

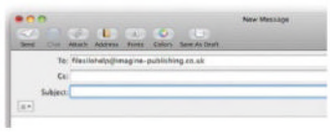
To access FileSilo, please visit **filesilo.co.uk/bks-798/**

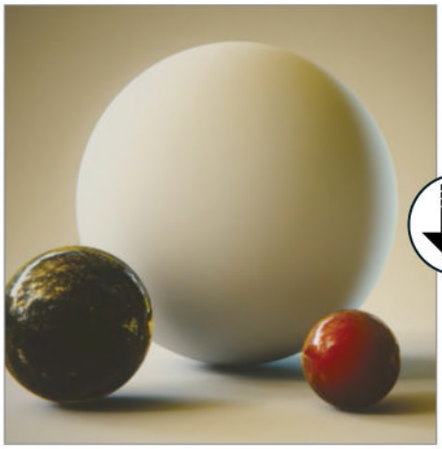
01 Follow the on-screen instructions to create an account with our secure FileSilo system, log in and unlock the bookazine by answering a simple question about it. You can then access the content for free at any time, and download it to your desktop.

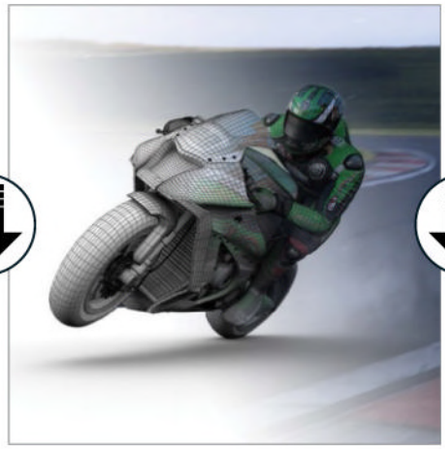
02 Once you have logged in, you are free to explore the wealth of content available on FileSilo, from great video tutorials and exclusive online guides to superb downloadable resources. And the more bookazines you purchase, the more your instantly accessible collection of digital content will grow.

03 You can access FileSilo on any desktop, tablet or smartphone device using any popular browser (such as Safari, Firefox or Google Chrome). However, we recommend that you use a desktop to download content, as you may not be able to download files to your phone or tablet.

04 If you have any problems with accessing content on FileSilo, or with the registration process, take a look at the FAQs online or email **filesilohelp@imagine-publishing.co.uk**.







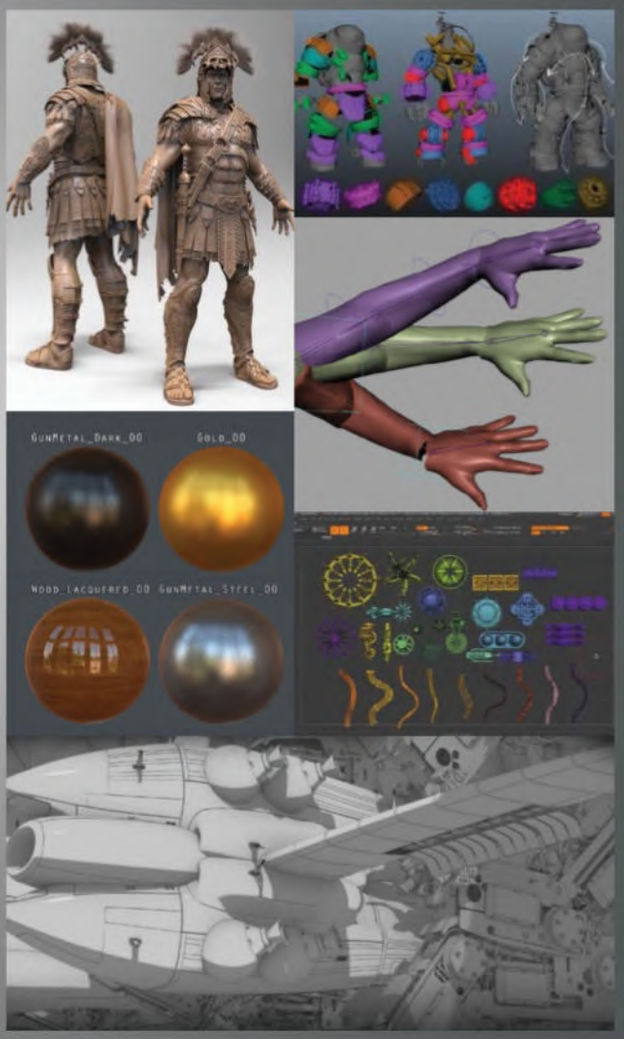
NEED HELP WITH THE TUTORIALS?

Having trouble with any of the techniques in this issue's tutorials? Don't know how to make the best use of your free resources? Want to have your work critiqued by those in the know? Then why not visit the 3D Artist and Imagine Bookazines Facebook page for all your questions, concerns and qualms? There is a friendly and active community of fellow 3D art and design enthusiasts waiting to help you out, as well as regular posts and updates from the team behind 3D Artist magazine. Like us today and start chatting, sharing your work, and getting involved with one of the brightest 3D art communities on the web!

facebook.com/ImagineBookazines
facebook.com/3DArtistMagazine

The 3D Art & Design ANNUAL

Take your 3D projects to the next level & transform the way you work



FREE DOWNLOADS

Free creative assets

The ultimate creative assets available to download online for all readers

- Tutorial files
- CGAxis models
- Essential resources
- Video tutorials