PBR Workflows

AR Design Handbook

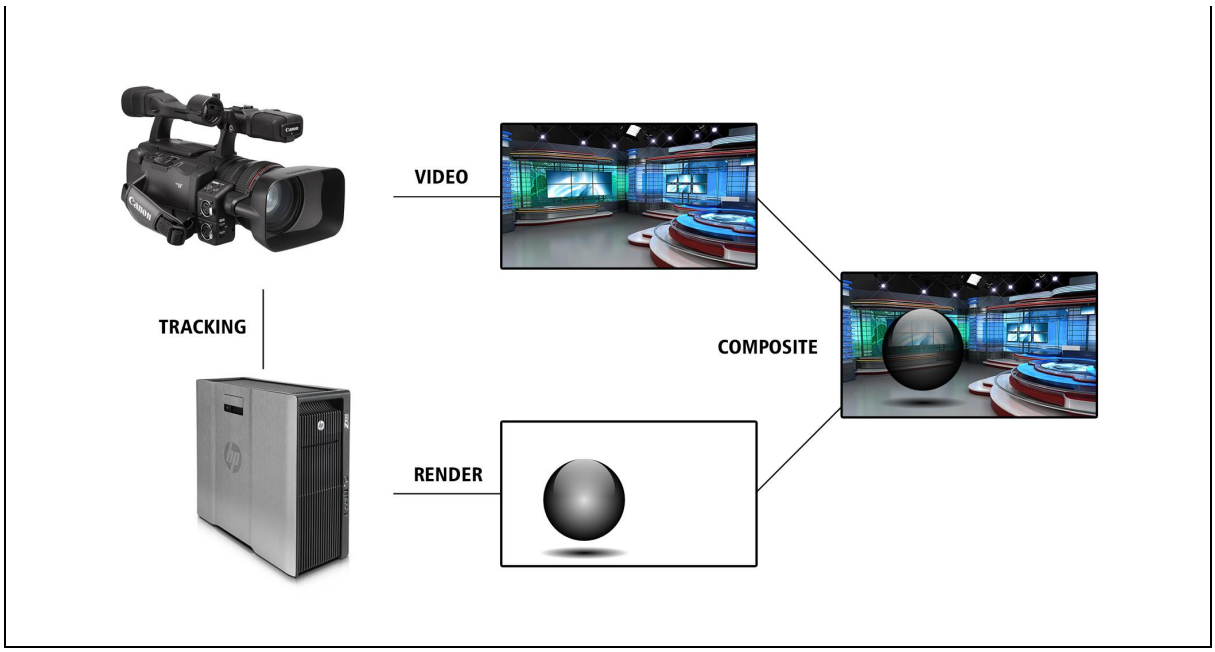
DATE	19/06/2015
PRINCIPAL AUTHOR	S Hickey;
SECONDARY AUTHORS	Darrell James
VERSION	
UPDATE	
MISC	Docs style applied Dec 2020

rtsoftware

Intro to Realtime

Explaining AR

Augmented Reality is an illusion. Using tracked cameras and a render engine, virtual objects can be made to look like they are placed in the real world (see image). This illusion requires certain constraints.



Explaining Realtime Constraints

Frame Rate - 60 renders per second

In post-production visual effects (VFX), it's okay to render a single frame in several minutes. In realtime, that's not okay. We need to render sixty frames every second.

How fast can you make a VFX render?

How fast? Maybe ten seconds per frame, right? If you turn all the settings down to minimum. Turn off shadows and raytracing? Turn off global illumination? Maybe you'll get one second per frame, right?

That's still 60 time too slow for realtime. So how do you get it sixty times faster than that? You find out what's slowing the renderer down and you remove it from the process. Maybe you take over some of the jobs that a VFX renderer normally does. That's the basis for realtime graphics: You are the one making things look pretty, not the renderer.

- (Note for Artists) In order to speed up the rendering time, artists must acknowledge that they will need to make compromises between aesthetics and efficiency. Often, effects that can be automated by the renderer in a VFX situation cannot be automated in an AR situation.

The differences between VFX and AR will be discussed throughout this document.

Performance Testing - The Artist's Responsibility

One of the biggest differences for VFX and AR artists, is that AR artists have a technical responsibility. Artists must be able to test the performance of their assets—they need to be able to put their work into the realtime engine and see how it affects the frame rate. If there is a problem, they need to know what steps they can take to fix it.

- (Note for Artists) Artists must understand this technical duty. It is vital for an AR artist to incorporate testing and optimisation into their workflow. If ignored, this is one of the biggest dangers to an AR project. Bad assets will lead to dropped frames, weird visual artefacts and unscalable projects.

- (Note for Artists) Senior Artists must understand the difficulties for VFX artists in adjusting to this new workflow.

Topology - How to Model for Realtime

> *"Topology refers to the structure of your mesh and the way the mesh flows around the surface and details of your model. Due to the complex nature of the subject, truly understanding it and grasping how to create clean, effective topology for your models, of all subjects, can not only be daunting but also very challenging." — Learning mesh topology collection*

In post-production visual effects (VFX), the lighting effects and reflections are handled using complex ray tracing renderers. These renderers are very forgiving—you can model an object without much thought to its composition and the renderer can overcome many of the problems in rendering the lighting effects.

However, in AR we must dispense with the complex VFX rendering tools. They take too long to render. Instead, we can use what is known as Per Pixel Shading to achieve these lighting effects. Per Pixel Shading is very fast and produces pretty good results. However, it is not very forgiving.

- **(Note for Artists)** Artists should understand that any corners they cut during the modelling process cannot be overcome by the renderer. They should understand that the renderer will show up any problems very clearly. This is a limitation of realtime. To overcome the problem, artists must learn to create models with perfect topology.

Models for realtime rendering should have a clean topology and conform to the current standard in 3D model making for realtime rendering:

- the model should have evenly sized quads arranged in clean loops with no t-vertices nor non-uniform faces;
- the objective should be to ensure that each vertex touches a maximum of four polygons;
- the occasional triangle is permitted in areas normally out of sight;
- the minimum number of polygons should be used to create the desired silhouette;

Polygon Count - How Many Triangles in a Scene?

While a high polygon count is rarely an issue in itself, the total number of triangles in a scene should probably sit underneath 2 million.

- (Note for Artists) Artists should understand that it is their individual responsibility to be aware of the number of triangles within a scene and how new additions to the scene will affect performance.

Transparency - Layers vs Raytracing

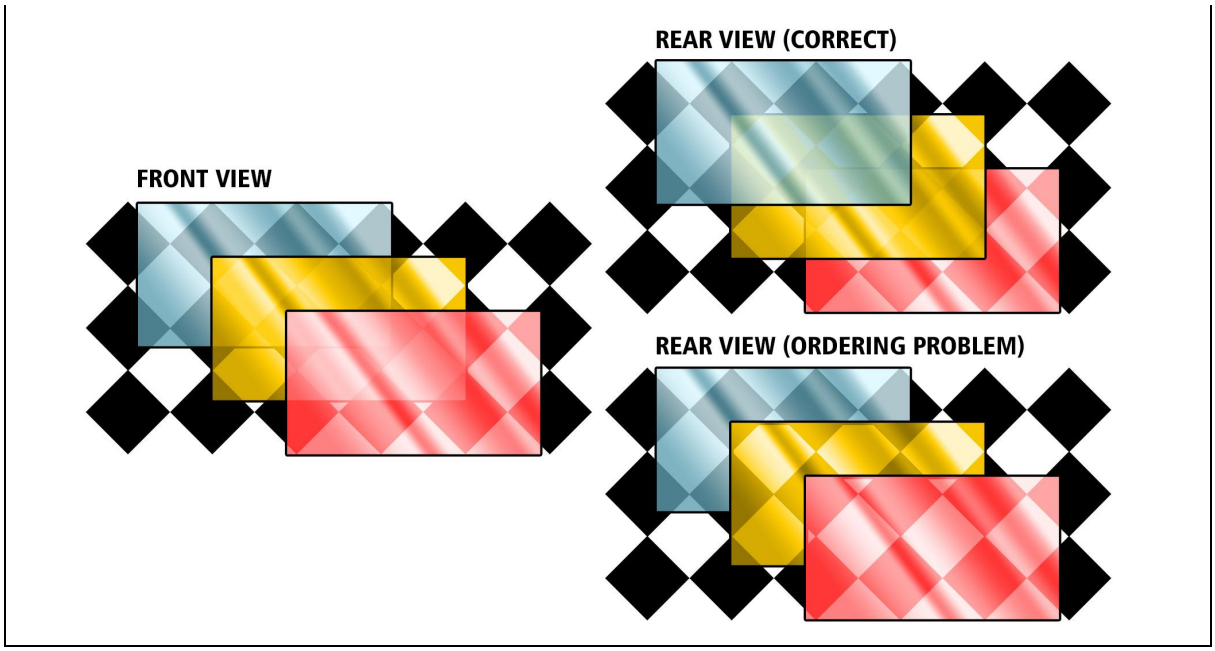
Transparency in a realtime engine is expensive and troublesome. Because we cannot raytrace transparency, we must blend transparent objects together differently—so we end up blending them together like Photoshop blends layers together: one layer blends on top of another layer which blends on top of another layer, etc.

This depends on knowing which layers blend on top of which.

In a static scene (a scene in which the camera and object ordering doesn't change), this is not a problem! You can safely say that one layer is in front of another which is front of another, etc. However, in a dynamic scene (a scene in which the location of the camera is unknown or the object ordering changes with animation), the ordering of layers may become problematic.

From the front view, everything looks right. The red is front of the yellow which is front of the blue.

If we were to take the camera behind the transparent objects (so we now look at the objects from the other side), we expect the blue to be in front of the yellow and the yellow to be in front of the red (see the top right image).



But in a realtime scene we have to rely on a fixed layering of the objects (we don't have enough time to work it out using ray tracing). So, the rendering engine is expecting the red to ***always*** *be in front of the yellow and the yellow to* ***always*** *be in front of the blue. So even though we look at the panels from the reverse side, the rendering engine is still expecting the red to be in front of the yellow and the yellow to be in front of the blue. So it draws the red first, then the yellow and then the blue. Which causes errors like you see in the bottom right image on the previous page. You see the problem? Not only does it put the red object first, it actually ends up cutting out bits of the yellow and blue!*

This is a difficult problem that can sometimes be overcome—albeit with some caveats.

- (Note for Artists) Artists must understand that transparency is a difficult problem to solve and, even when implemented correctly it can end up putting limitations on the movements of the cameras during Tx.

Explaining Keying Constraints

The difference between what people call VR and what we are calling AR, is that VR uses a green screen. With a green screen, you can replace the green with a virtual environment. In this way, you can have presenters appear to be in front of the virtual environment.

Essentially, you replace the green area with a virtual object.

But without a green screen, there is nothing to replace. So the only way you can add a virtual object into the scene is to put it in front of everything. This means you can't put a virtual object behind the presenter. Every object needs to be in the foreground. **Without a green screen, the presenter can never walk in front of an AR object.**

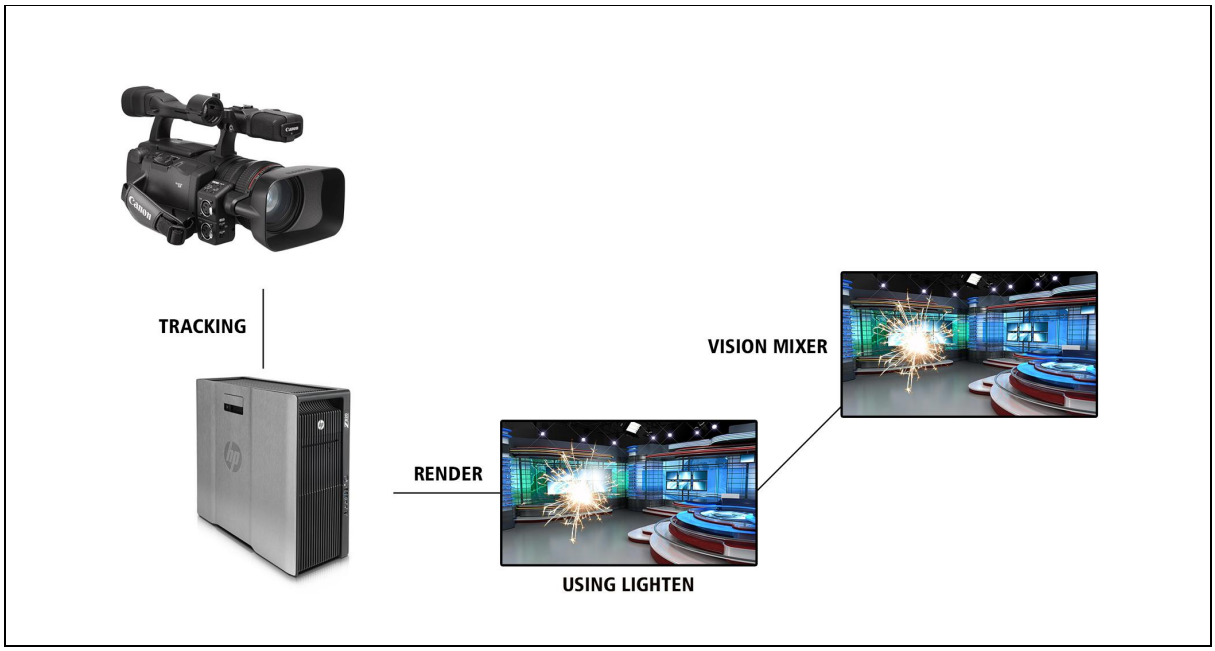
- ❏ Artists must understand that without a green screen, the virtual objects must always be in the foreground and that presenters can't walk in front of them.

Explaining Transfer Constraints

There are two ways to combine AR graphics into the video feed. You either get the render engine to do it or you do it in the vision mixer.

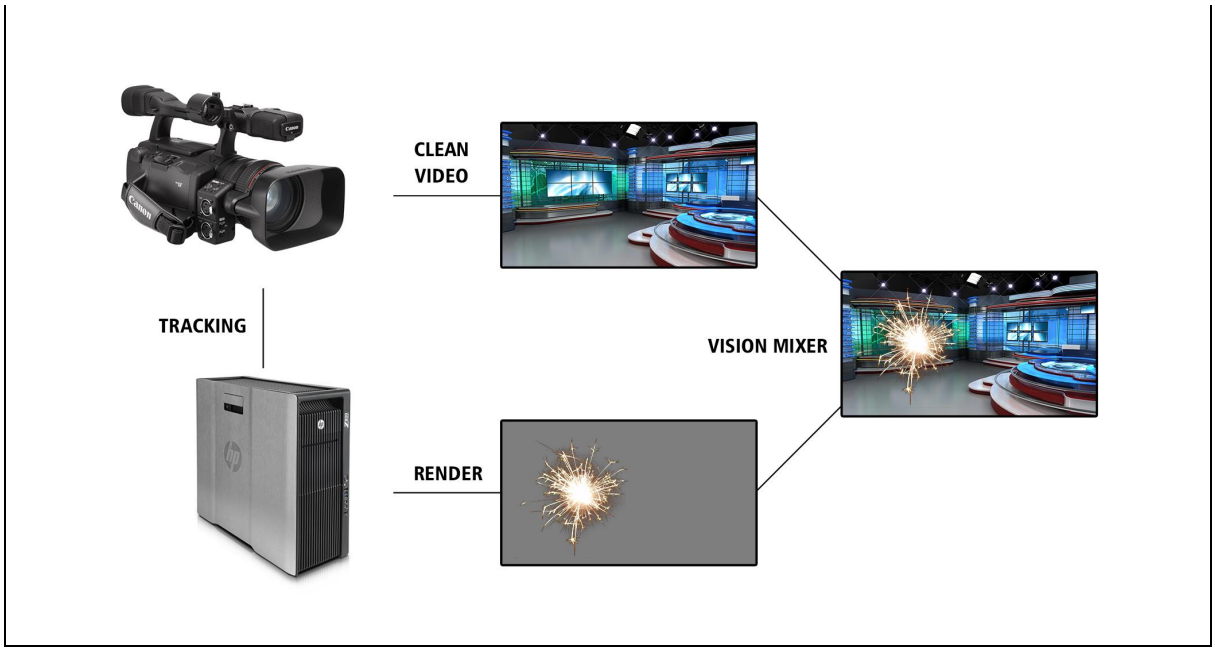
Render Engine Compositing

If you pipe the clean video into the engine, combine the graphics and send a dirty feed to the vision mixer, you can take advantage of all the layer blending modes for objects in your scene. For example, if you wanted to create sparks in your scene, ideally you would like those sparks to blend into the video layer using a mode such as Screen, Overlay or Linear Dodge (Add). If you use the engine to process the video, you can take advantage of these kinds of effects.



Vision Mixer Compositing

However, it is common practice to mix the AR onto the video at the vision mixer. In this case, the engine outputs a fill and a key channel and these are blended with the AR with a simple mix. With this workflow, the rendering engine does not get access to the video feed, so it cannot meaningfully apply those effects.



- (Note for Artists) Artist must understand that if the workflow uses the Vision Mixer, they will not have advanced blending modes available. This should have an impact on design decisions that rely on advanced blending modes, such as particle effects.

Explaining Tracking Constraints

VFX artists may be familiar with post-production camera tracking techniques. These are applied on top of a piece of video footage and use software algorithms to try to estimate the camera location during the shot. This can take some time, depending on the software you are using.

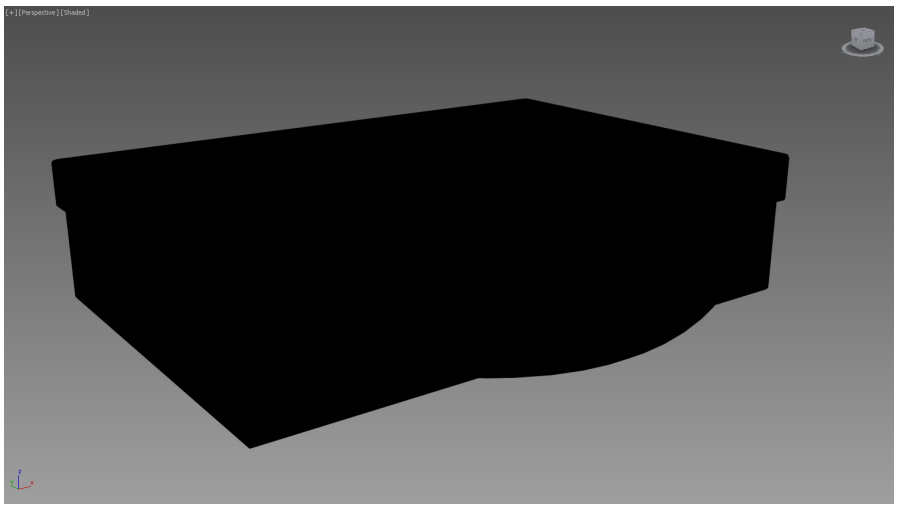
In broadcast AR, we need to work this out live and feed it directly into the render engine as least as fast as the render engine itself. If there are any errors in the tracking system, you may see the AR object drift.

Even the best tracking technology in the world, in the best conditions, is only accurate to around 5 mm. Often, you'll see drift of around 4 cm or more! This means that if you are aiming to line up AR objects with real objects in a set, you will notice the drift pretty badly.

- (Note for Artists) Artists should understand that lining up an AR object with a real world object is difficult and will draw attention to any inaccuracies in the tracking system. It is strongly advised to keep AR objects in areas which don't draw attention to drift.

3D Modelling and Texturing Overview

Modelling for AR - The Silhouette

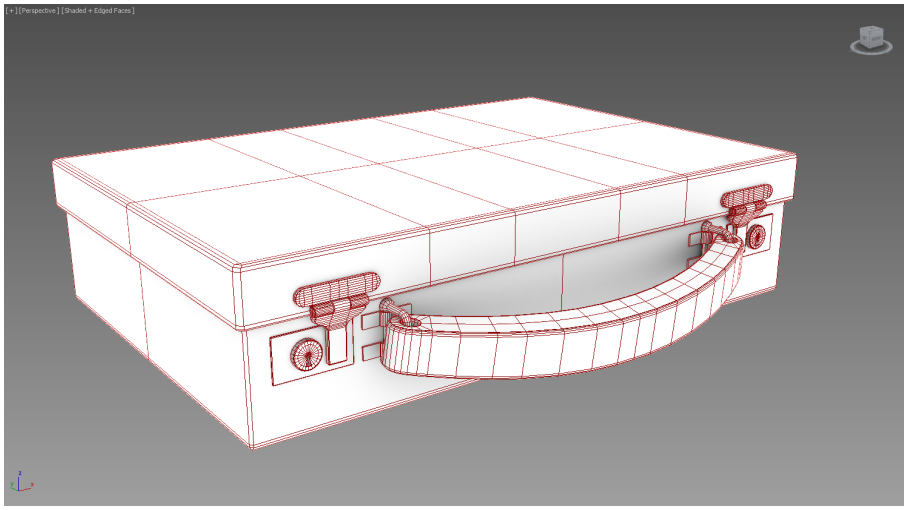


When designing models for AR, try to think of the silhouette of the object. Only model enough detail so that when you view the object from various angles, the silhouette is preserved.

High frequency detail should not be modelled (for example, on a briefcase you needn't model the stitching, leather pattern or screws). These kinds of detail should be recreated using textures, not geometry.

- ❏ **(Note for Artists)** Artists should understand the value of reducing a model down to its silhouette.

Modelling with Good Topology - Overview



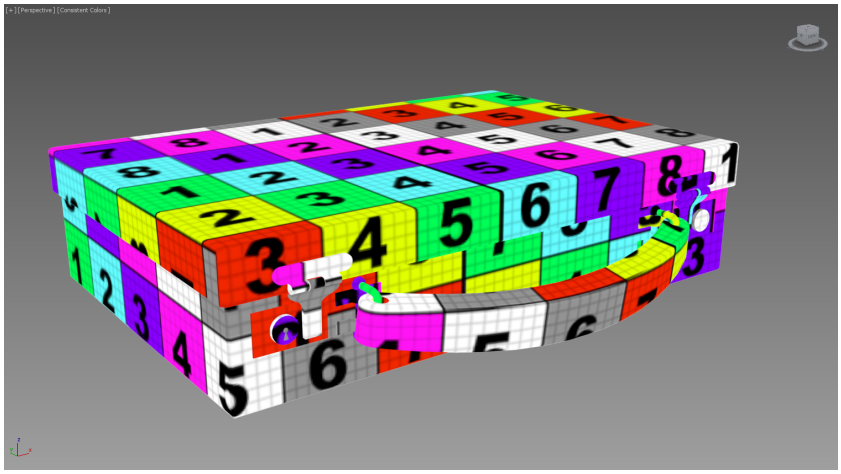
When modelling for realtime, start by blocking out the major shapes and adding detail as you go. Only model using quads.

- **(Note for Artists)** Artists, if you have to choose between a triangle or an extra loop of polygons, add the loop of polygons. Or move the triangle to where it won't be seen.

Keep the quads as even as you can. Of course, some polygons will need to be narrower than others, but try to make sure that's consistent around the loop, Make sure you don't end up with non-planar polygons or t-vertices.

- **(Note for Artists)** Artists should understand t-vertices and non-uniform faces may cause problems in realtime rendering that might not be immediately apparent during the modelling process.

UV Mapping for AR - Overview



Since we're trying to keep rendering costs to a minimum for realtime, we have to generate a lot of the surface detail and shading manually in a texture map. In order to begin that process, you'll need a good UVW map.

While it's okay to automatically unwrap UW maps for simple objects, be aware that an automatic unwrap can cause problems for more advanced AR assets. Particularly, an automatic unwrap …

- is not suitable for projecting a sculpt.
- is not suitable for a Photoshop painted asset.
- is not suitable when using ultra high resolution textures (it tends to waste texture space, which becomes expensive at higher resolutions).

However, if you're going to paint your asset using BodyPaint or Substance Painter, or you're going to bake out an ambient occlusion pass, it might be okay to perform an auto unwrap.

- **(Note for Artists)** Artists should be aware that automated unwrapping is not suitable in some situations and that the better they pack their UVs, the better the textures will look inside the engine.
- **(Note for Artists)** Furthermore, when unwrapping an object, it is absolutely vital that the UV faces are normalised according to their physical size, otherwise texture

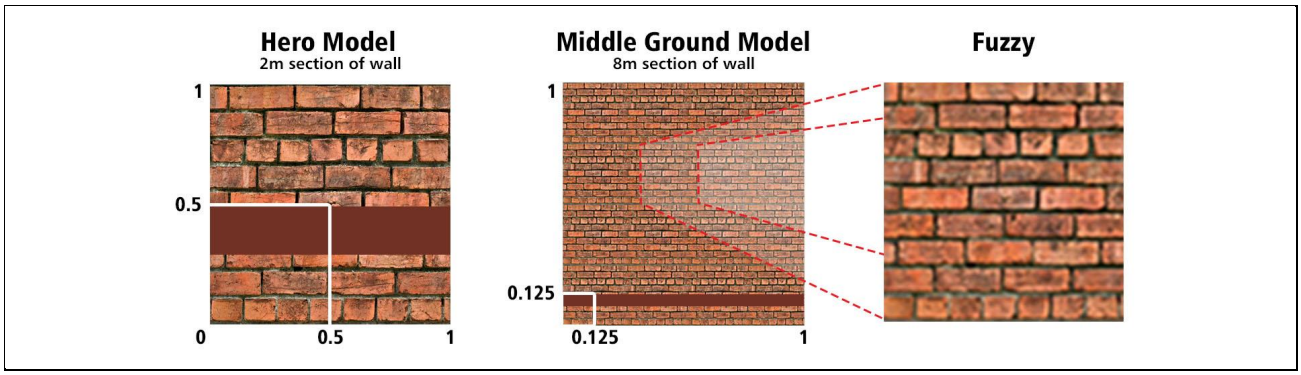
painting will become unpredictable, even with tools like BodyPaint and Substance.

- ❏ **(Note for Artists)** For soft edges, joined seams will give you more support when you're painting in a 3D viewport.
- ❏ **(Note for Artists)** Pixelation is more likely to occur on your maps if you don't unwrap efficiently.

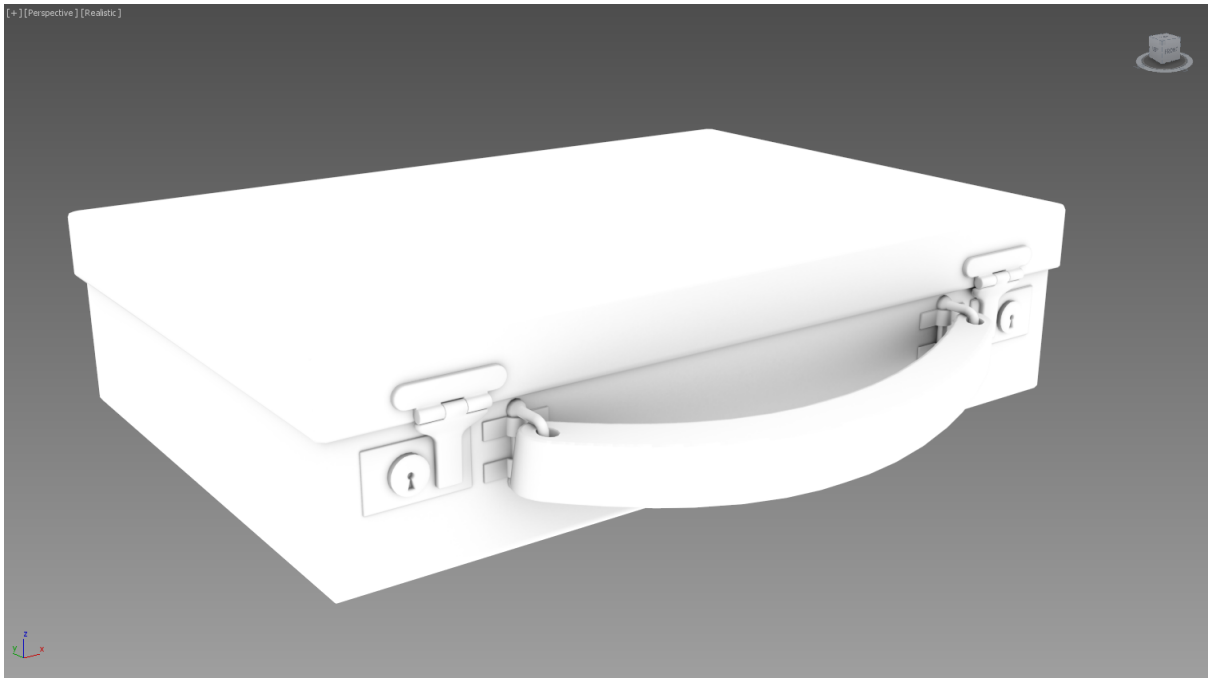
Advanced UV Unwrapping - Texture Density for UHD

When unwrapping an extreme foreground model (sometimes called a "hero" model) that takes up most of the view, you need to be smart about unwrapping texture density.

- Hero models are models that will be used in the extreme foreground.
- To future proof assets for UHD, artists should unwrap models to 0.5 UV per meter of physical space for hero models (which equates to 2048 px within a 4K texture).
    - As a rule of thumb, if a surface is larger than 4m, break up the surface into separate, 4m objects before unwrapping.
- For other assets, unwrap to 0.125 UV per meter.



Ambient Occlusion (AO) - Overview



Having modelled an unwrapped your UVs, the model is ready for texturing. To start, render an ambient occlusion (AO) map. The size of the map should be dictated by the importance of the object. If it's a hero object (right up close to the camera), start with a 4K map to see how it looks. You can always downsize later if there's a performance issue.

If the object is going to stay in one place without any animation, bake the AO map in-situ so that it picks up shadowing from nearby elements.

- ❏ **(Note for Artists)** Artists should be aware that if the object is to going to animate, they will need to consider baking the AO map in isolation.

Having rendered the AO map, this can be used as a guide for painting further detail onto the mesh.

Texture Sizes and Basic Painting - Overview

As indicated by the Advanced UV Unwrapping - Texture Density for UHD section above, your map sizes vary depending on the importance of the object and your desired texture density. For starters, work with 4K images.

The size of the textures will vary from object to object. The less prominent the object, the lower the resolution. For basic AR assets using the default shader in tOG, you can paint on top of the AO map in Photoshop. Paint your shadows, colour and some soft highlights into the diffuse map.

> *"Imagine that most of your high frequency lighting will come from your texture, not from the light sources within the scene. Use reference."*

- (Note for Artists) Artists should use 4K images for prominent AR elements and lower resolution images for less prominent elements.

Simple AR Effects Overview

This section introduces solutions appropriate for simple realtime effects.

Fresnel Reflections - Using Shaders for Reflections

Though we can't expect the realtime engine to ray trace reflections from objects in the scene, we can fudge a decent reflection of the environment using a Fresnel shader.

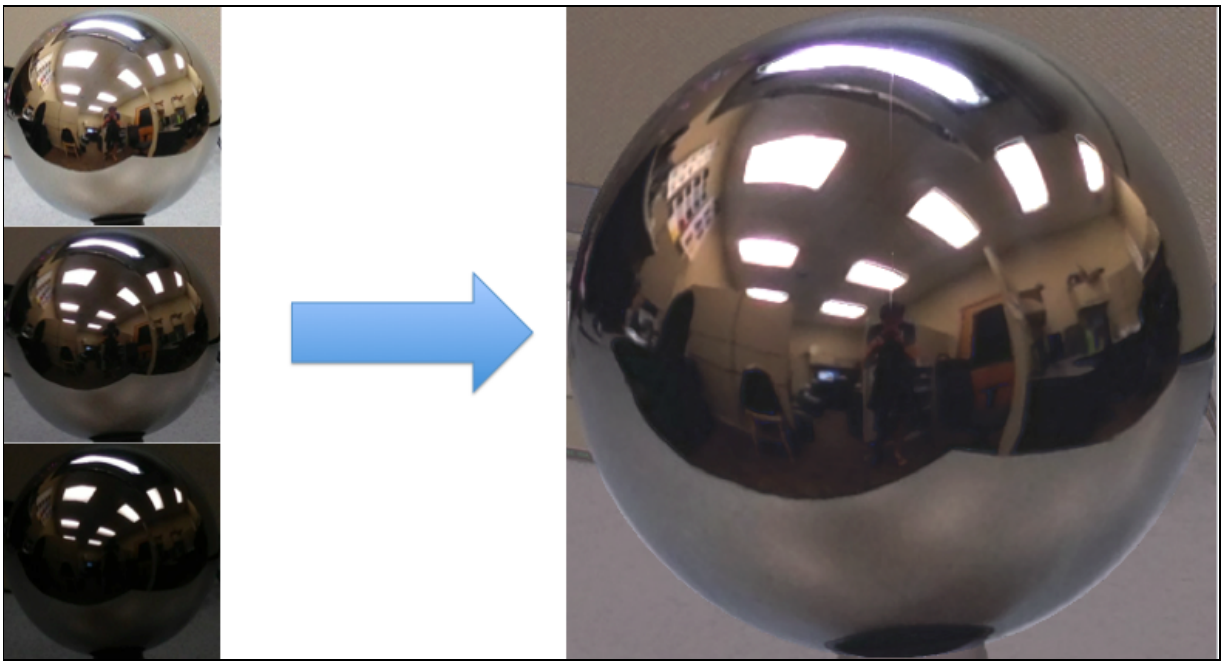
- **(Note for Artists)** Artists should understand that Fresnel Reflections rely on a clean topology. If you don't have time to create a clean topology for your mesh, don't use Fresnel Reflections.

If you want the object to have a consistent amount of reflection across the surface, you don't need a UV map. Simply add the shader and the environment map.

But if you want to vary the reflection across the surface (to simulate reflectivity or ambient occlusion), consider adding a specular map to the object.

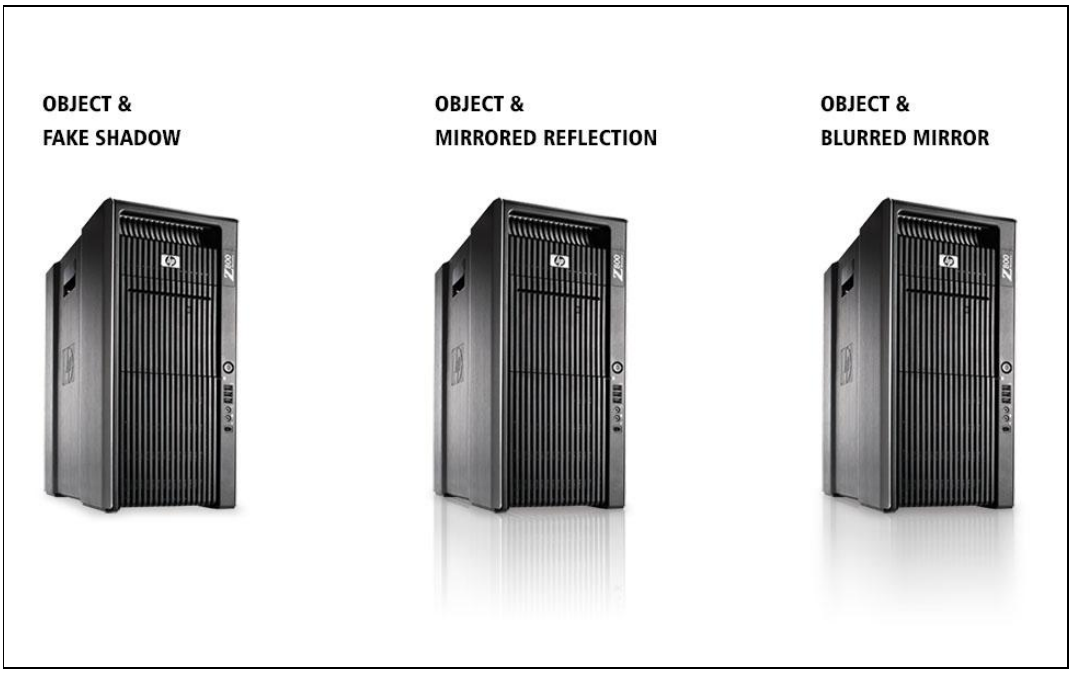
Environment Map - Something for Objects to Reflect

It is possible (and advised) to light an asset in realtime using an environment map. This process yields good looking results, even in realtime.



- Recommend using a 360 camera like Ricoh Theta to create spherical maps …
    a. Shoot three exposures (-2, 0, +2).
    b. Use Photoshop to generate two more exposures using the Exposure adjustment (total exposures: -4, -2, 0, +2, +4).
    c. Without bespoke hardware, you will need a tripod and a DSLR camera with a wide lens, allowing half an hour to shoot each environment.
- Combine exposures with Import to HDR Pro.
- Non HDR images will require levels adjustments to push contrast.

Simple Reflections - Cheating Flat Reflections



For simple reflections in flat surfaces, it is possible to create satisfactory results just by mirroring the objects applying a gradient mask within tOG Edit. Furthermore, the mirrored surface could be then blurred using an additional layer effect within tOG Edit (though this blurring can be expensive).

- (Note for Artists) Artists should understand that dynamic animation is possible with this effect (animated elements will be reflected on the flat surface).
- (Note for Artists) Artists should understand that if the reflected elements contain transparency, the ordering of the layers becomes important and potentially problematic.

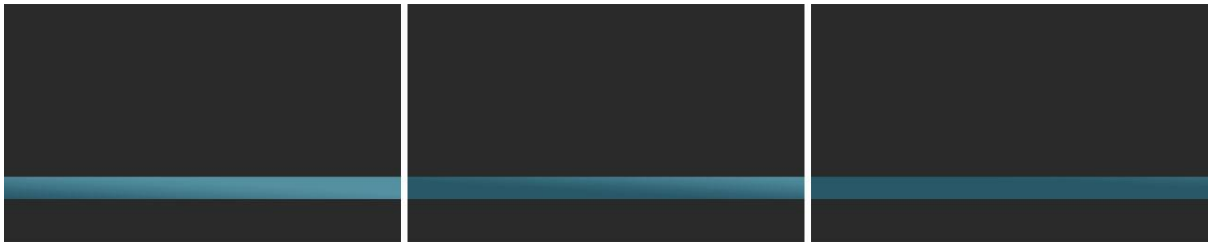
Images & Sequences - Simple Animations

You can cheat a lot of effects in realtime using simple images. Simply stick a PNG with transparency onto a plane and place it into your scene within tOG.

Still images

For example, to create a flare that travels along the edge of a lower third, simply animate the position and scale of an image of a flare.

Tiled image loops



If you have a tiled image, you might want to translate the tiling within tOG Edit. This allows you to add subtle animations to surfaces, like animating a gradient to simulate a specific lighting effect.

Exported 2D animations with alpha

Lots of lighting effects and particle effects can be created in a third party application and output as a sequence of images. If you want to add a beam of light, fire, lightning, sparks or simple glowing particles, you might want to try using an image sequence, especially for very quick sequences or background elements.

- **(Note for Artists)** The important thing to remember here is that your blending modes are limited by the video pipeline (see Transfer Constraints, above). So you may have to be clever about your alpha mask.

Additionally, be aware that these are 2D effects.

- **(Note for Artists)** As soon as you put them into 3D, they will either be mapped onto a 2D surface (so you should think about what it's going to look like from acute angles) or a 3D surface (so you should think about how the 3D model will distort the 2D image sequence).

Complex AR Effects - Overview

Ordering Objects for Transparency

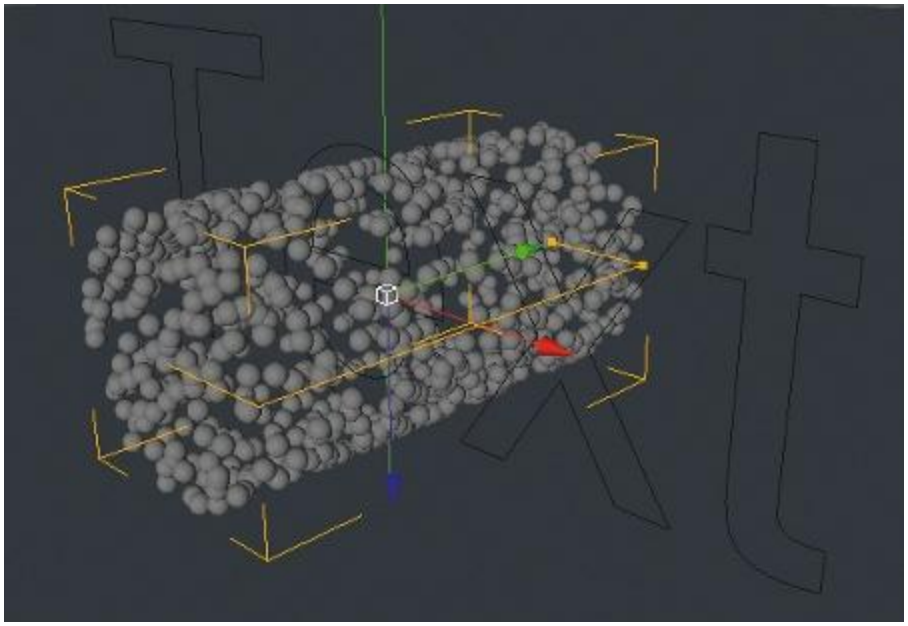
Transparency is difficult to overcome and expensive. Solutions are rarely straightforward and there is normally a compromise.

1. Know your camera shots.
2. Order your objects so that they work in the most common cases.
3. For uncommon cases, you may have to duplicate and flip the normals of problematic meshes. This should be done in consultation with RT Software.
4. Also, consider removing transparency in conflicted scenarios.

Baked Dynamics - Exporting Animations from Cinema

With plugins like NitroBake, it is possible to bake out the animation of dynamic elements from Cinema 4D (for example).

However, the individual objects will not self-shadow or self-reflect. This would be most useful for self-illuminated elements like a video screen, dark or reflective objects or particles (if transfer modes permit).



Baked Mesh Sequences - Exporting Animations from Cinema

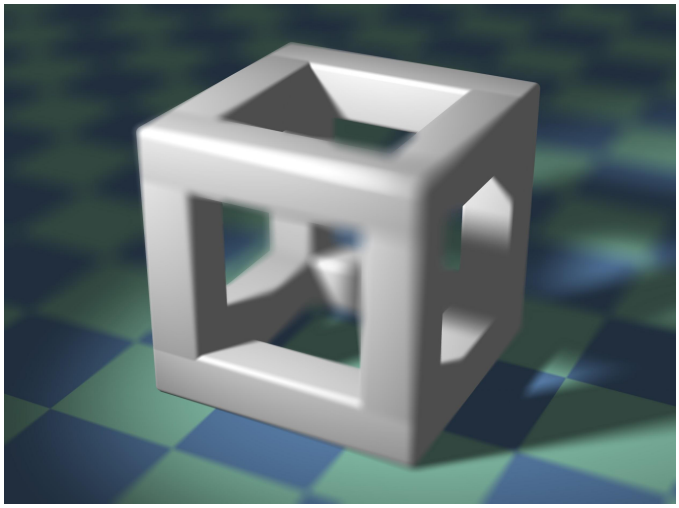
With plugins like Riptide Pro, it is possible to take a complex mesh deformation and bake out a sequence of meshes.

As with baked dynamics, there would be limited shading effects, though using the Fresnel Shader could yield some interesting results. Perhaps the effect could be used to manifest the 3D transitional elements between two correctly shaded states.

Depth of Field - Probably Not Worth the Cost

Difficult to match effectively to a real camera.

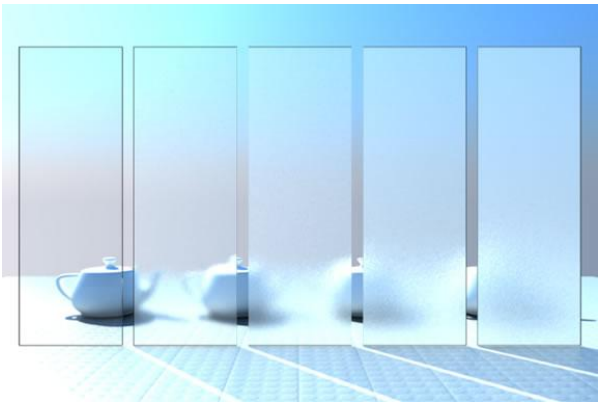
- Expensive to implement.



Refraction - Another Kind of Transparency

You can achieve some nice results in realtime, especially for glass panes and frosted glass effects.

- (Note for Artists) Be careful about ordering your objects.



Sculpting - Adding High Poly Detail to a Low Poly Mesh

Using tools like ZBrush, Bodypaint or MudBox it is possible to create intricate detail in a high poly mesh and then map that detail onto the low poly mesh with a normal map.

Physically Based Rendering Shader - Adding Realism

Realtime Physically Based (PBR) Shading, with supported workflows (Quixel Suite, Substance, Marmoset) enable realtime artists to make fewer compromises. It is worth investing some time learning Substance Painter in order to speed up the workflow for texture artists.

Chapter 2: Avoiding Gimmicks

This chapter assumes you know already know how to create compelling designs.

So this is about avoiding gimmicks.

When we're learning a new platform, it's tempting to try new things (just because we can). We're going to offer one possible solution to that problem.

Design by Objectives

This solution puts objectives at the center of design. By referring to these objectives throughout the build, we ensure graphics have purpose.

It's only when graphics don't have purpose that they become gimmicks.

Here's what it boils down to:

- Describe our **objective**(s).
- Figure out the **core** message, including …
    - The conclusion we are hoping the viewer to reach.
    - The core piece of data or information we'll explore.
- Describe our **constraints**.
- Tell the **story**. Beginning. Middle. End.
- **Prototype**.
- Review. **Review**. Review.

Describing Objectives

State the objective.

Break it into pieces (components). Propose how we can meet the objective (assumption) and how we can test whether our assumptions were correct (test).

- Objective -> Components
- Components -> Assumptions
- Assumptions -> Tests

Example

Here's a case study from Estonia. We had a couple of weeks to create an AR set extension for a children's show in Estonia.

Objective

Provide a **variable** AR set extension that would present a **credible**, **child friendly**, virtual environment.

Components

- Variable
- Credible
- Child Friendly

Assumptions and Tests

- **Variable**: We assume that if we provide two set extensions, they'll understand the possibilities for variation.
    - [Check with client if two sets provides enough variation.]
- **Credible**: We assume that if we add depth and incidental motion to each set, credibility will improve during crane moves.
    - [Test greybox in demo room/studio box with a camera dolly/track.]

- Child Friendly: We assume that if we include simple childhood symbols, the scenes will be appropriate for the show.
    - [Review deliverables with client.]

What’s the Core?

Don’t start creating a solution yet. First, figure out the core of the story.

Describe the Conclusion

What conclusion do we want the viewer to reach?

For example, with the set extension we wanted the children to conclude that the studio was located in a wondrous destination. But for a head-to-head sports graphic, we may want the viewer to recognise the underdog competitor. Or, differently, to learn what to look out for during an upcoming match.

Core

What’s the core medium (stats, locations, mugshots)?

For example, with the set extension there were no stats or mugshots, the core medium was the set itself. But for a head-to-head sports graphic, the core medium would probably use stats.

Define Constraints

Define the real, **physical constraints** that will limit our solutions. Including:

- Set dimensions
- Camera editorial
- Lighting conditions

Define the **visual constraints** that will limit the design aesthetic. Including:

- Brand

Lastly, treat the **objective as a constraint!** Objective constraints will help us to remember what we're aiming for. Keeps us in line.

What’s the Story?

Now it’s time to create the solution.

First, look at the conclusion and the core. These should tell you what you’re aiming for. Now look at the constraints. These should help you understand how to attack the problem.

Visualising data (whether it’s a mug shot or a match statistic) is like telling a story. You’ll need to come up with a beginning, middle and end.

Beginning

Introduce the graphic like it’s never been seen before. Show how it works. Acclimatise the viewer.

You’ll want to forecast what is about to happen. Maybe it’s a slow pan to reveal an AR object while the presenter talks about it. Or maybe it’s a shadow on the floor before an object descends from above. Or maybe it’s a flash of light to draw the attention to a particular area of the shot.

It’s all about making the viewer comfortable. Giving a little warning. Sowing some expectation. What you don’t want is for the viewer to be surprised by the graphic and confused why it’s there. So, in your storyboard, figure out a way to introduce the graphic. This is the beginning.

Middle

Once the viewer has acclimatised to the presence of the graphic, deliver the core. You should be trying to clarify what the graphic is for as clearly and as quickly as possible. This is where you bring on the stats (or images, or formation elements, etc.), whatever tells the story. The moment you’ve delivered the core, the viewer will try to get some meaning so make it easy. Keep it simple.

End

Once the core is in place and it's completely clear what the graphic is representing, allow the viewer to come to a conclusion. Or draw attention to it.

You ought to be thinking of the read (where the viewer will be looking) and how to draw attention toward or away from specific areas. You can use this to momentarily highlight a specific part of the graphic, for example. The idea is to give the viewer just enough information for them to come to a conclusion about the data that you've represented.

Story Example - Team Underdog

Let's say you've got a football match between two unfamiliar teams in a knockout cup and you want the viewer to get emotionally involved. You reckon that if you can highlight the underdog, you might be able to achieve that result.

So, you get started on the introductory graphic for the teams. The conclusion you want the viewer to make is that one of the teams is outmatched despite their struggle through the playoffs. You've decided to deliver these conclusions using stats and match history (these make up the core).

Let's break this into two sequential stories within the same graphic: first present the struggle through the playoffs, then present the head-to-head comparison.

Story 1 - Knockout Map - Beginning

The show starts with a wide shot with the presenter in the middle of the set, with a team badge on either side. *This introduces the AR elements as part of the show and gives the viewer time to acclimatise to the situation.*

Story 1 - Knockout Map - Middle

The presenter announces the teams, talking about their history so far in the competition. As he/she talks, each of the two badges animates to reveal a glassy plinth titled "Knockout Map."

The Knockout Map shows all of the other teams that they've knocked out of the

competition so far. It should look like a knockout tree—so the more teams you've knocked out the longer the tree. *This allows a comparison between the team's history. The team that has been in the competition the longest will have a longer tree. This is the core. Give the viewers some time to read.*

Story 1 - Knockout Map - End

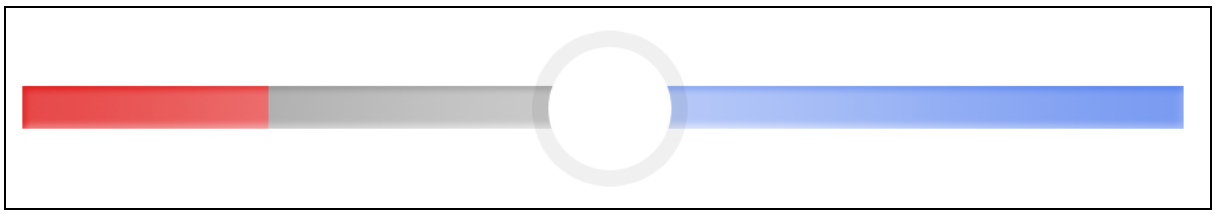
We highlight, one by one, the teams knocked out (simultaneously in each Knockout Map). *This draws attention to the disparity (if any) between the teams' journeys through the competition.*

Story 2 - Versus Comparison - Beginning

There is a subtle flash and the two badges are suddenly connected by a curved silver bar. *This sets up the area for the head to head comparison.*

Story 2 - Versus Comparison - Middle

The title of the bar is revealed to say "tonight's prediction" with a victory cup in the center. The bar starts to fill up with one team colour on the left and the other team colour on the right. The colours flow toward the center.



The better performing team fills up its half of the bar with its team colour. The underdog fills a comparatively (based on stats) smaller portion of its half of the bar with its team colour. *Now the viewer has all the information it needs to draw the conclusion whether one team is more or less likely to win.*

Story 3 - Versus Comparison - End

Once the colours settle and the viewer understand what the colours mean, the victory cup is filled with the colour of the better performing team. *The viewer has had time to figure out the predicted winner—this reinforces that conclusion.*

Prototypes & Composition

Before you start work on delivering the solution, it's worth communicating the physical and visual constraints. These are typical methods used in realtime for that purpose.

Greybox

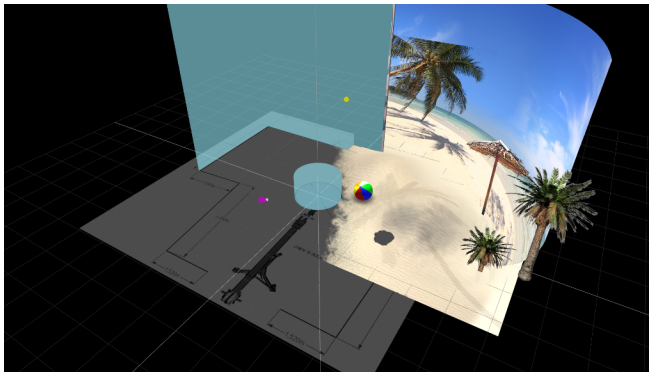
This helps communicate physical constraints: Placeholder AR objects are assembled within the live environment to help make compositional decisions or instruct the editorial. Often artists use primitive geometric shapes with a grey shader—hence the term.

Studio Box

Helps communicate physical and visual constraints. The studio is recreated as a simple virtual object with textures appropriate to a specific lighting scenario, for previewing.

Reference

Helps communicate visual constraints. A scrapbook of visual reference is used to communicate initial design ideas and inform conversations with stakeholders.



Prototypes & Composition - Example

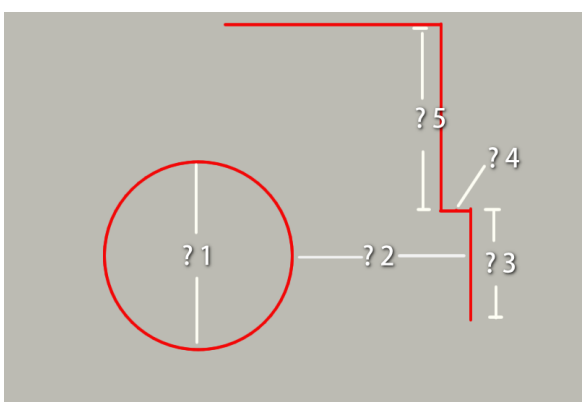
Review the Prototype

It's a good idea to review your prototypes. This means taking your prototypes into the studio to test in-situ, even when they're not finished.

Anyone can comment on a prototype. So show it to people. See if they understand your story. Do they get it?

Next, check the objectives. You might want to check your prototypes against your

objectives. Any problems you find there will save you lots of time down the line.



For example, with the set extensions in Estonia, we used the prototypes to check variability and credibility—the two sets were enough to communicate variability but we found the

camera tracks needed parallax layering to sell the snowy landscape.

Catch technical problems

Look from every angle. You'll probably find a rendering issue or a technical issue that needs to be resolved. The earlier you catch these the better, as sometimes they involve software tweaks to fix.

For example, with the beach set extension, we found problems with the aspect ratio of the beach scene videos.

AR Review

Late reviews must be done in-situ. That's not a suggestion. Let's be clear: If you don't want to review in the studio, you will absolutely get surprised during Tx. This ***will*** happen**.** So don't compromise on this, leave time for reviews in-situ and leave time for rework.

Review the Story. Is the story clear? Could it be clearer?

Review Objectives. Are we passing our tests? Are we meeting our objectives. Step back. Be particular.

For example, in the last phase of the snowy forest we realised we'd inadvertently put a murderous axe in the center of a pool of light. Not very child friendly.

Review Editorial

Check the known shots. Look for little errors.

Chapter 3: Topology

Modelling Tips

When modelling for realtime, you'll need to plan carefully. Use reference to inform the plan.

Think about the lowest poly count that will preserve your silhouette. Also think about bevels, extrusions and making holes. These are the kinds of procedures that could raise topological issues down the line.

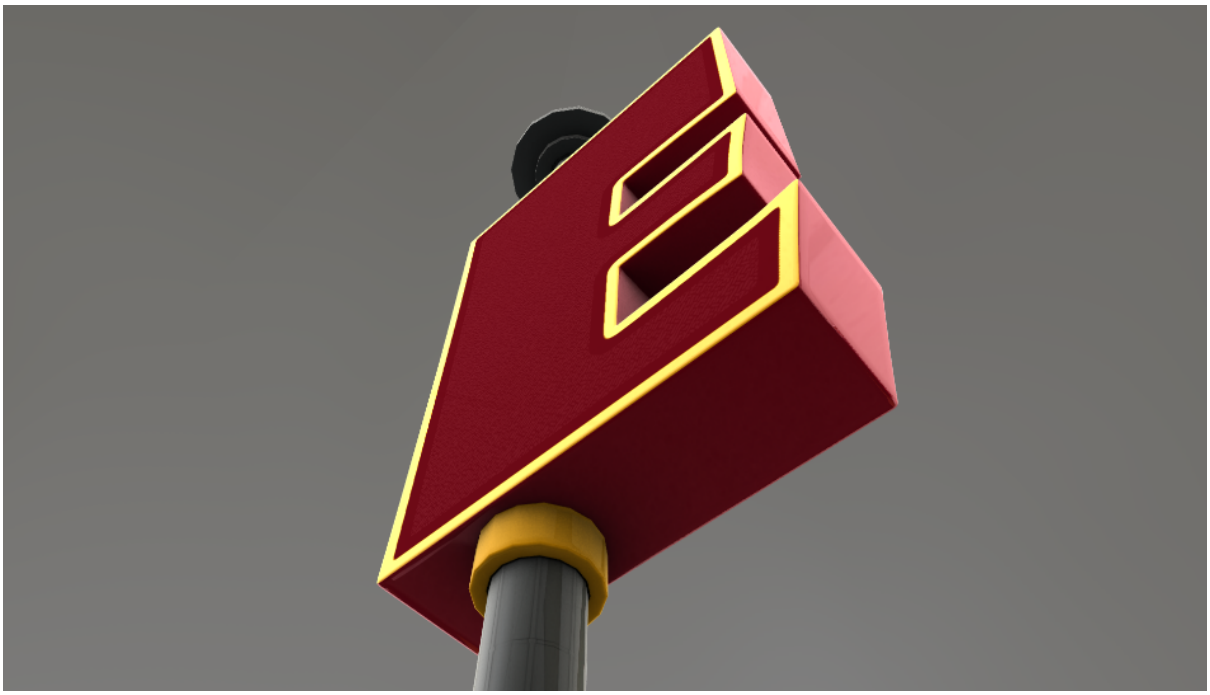
Always Model with Loops

To explore modelling with loops, let's imagine we're modelling a letter in Cinema 4D.

For letters, start with a spline.

1. Reduce the intermediate points.
2. Knife the spline so there's enough verts to create quads later.
3. Extrude the spline into a 3D shape.
4. Knife it into quads.
5. Turn off phong break angles.
6. Add inner extrusions to the top and bottom surfaces.
7. Knife in support loops.

In this way, your model will have quads along the spine and you'll reduce the frequency of poles on your edges.



How to Cut Holes with Loops (no Booles)

Never cut a hole using a Boole. Instead, create extra geometry with loops and form the geometry into a circle with a script.

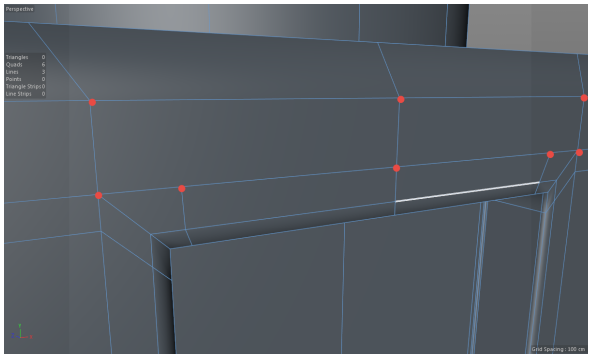
1. Add five* horizontal and five* vertical loop cuts to the surface using the knife tool.
2. Use the Points to Circle script to arrange them nicely. http://forums.cgsociety.org/attachment.php?attachmentid=133918
3. Delete the inner polys. Fill up the hole again.
4. Inner Extrude it twice, to avoid problematic poles.
5. Extrude it into the surface.

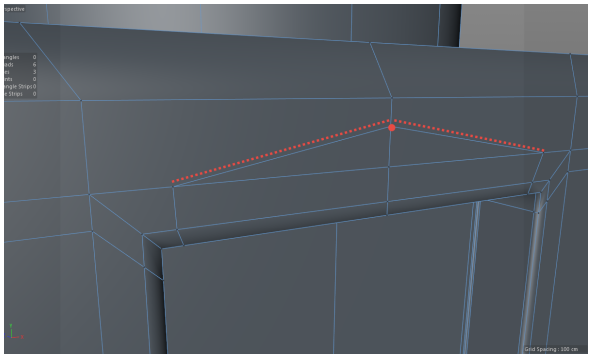
* If you’re going to use Subdivision, you can get away with three horizontal/vertical cuts. Which is easier to manage. :)

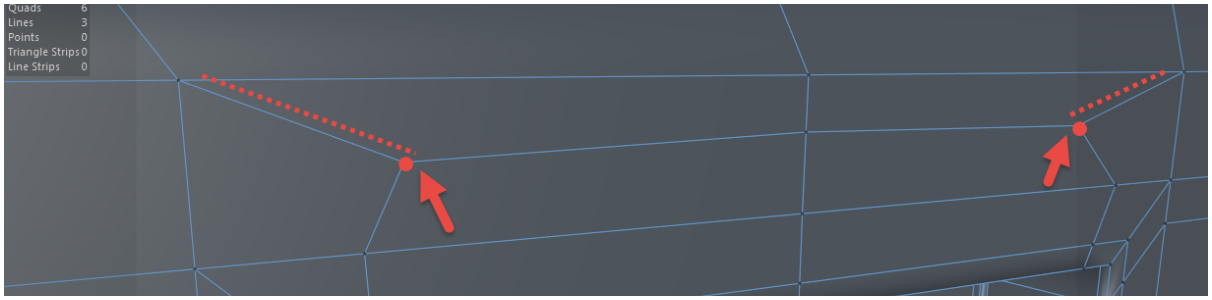
Redirecting Problematic Loops

Sometimes you have to create loops that will badly affect other areas you've already modelled (particularly with holes). Let's assume you want to add an extrusion onto a surface but if you add support loops to that extrusion it will mess up your hole.

1. Inner Extrude on some polys near a hole.
2. Inner Extrude again, to give yourself more breathing room.
3. Extrude it to give you your desired appearance.
4. Add support loops.
5. For problematic support loops, restrict the knife to the extruded face.
6. Redirect N-Gons (see below).






Quads 6
Lines 3
Points 0
Triangle Strips 0
Line Strips 0

A Guide to Fixing Topology

A few tips to help solving topology issues. First things first:

Is it broken?

1. Will this mesh have any dynamic lighting (including reflections)?
    a. If no, stop worrying about it. If yes, move to step 2.
2. Will you be subdividing the mesh?
    a. If yes, do you see any pinching when you subdivide the mesh?
        - If yes, fix the topology then move on to step 3.
    b. If no, move to step 3.
3. Export the mesh without any UVs.
    a. Apply an appropriate reflective fill layer in Substance.
    b. If the reflections behave, you're done.
    c. If the reflections are misbehaving, fix the topology in your 3D package and test again.

Visual Artefacts

If the lighting looks crappy on your model, it could be a topological issue. We can call those issues "artefacts." Once you find a visual artefact, you can start looking for the cause.

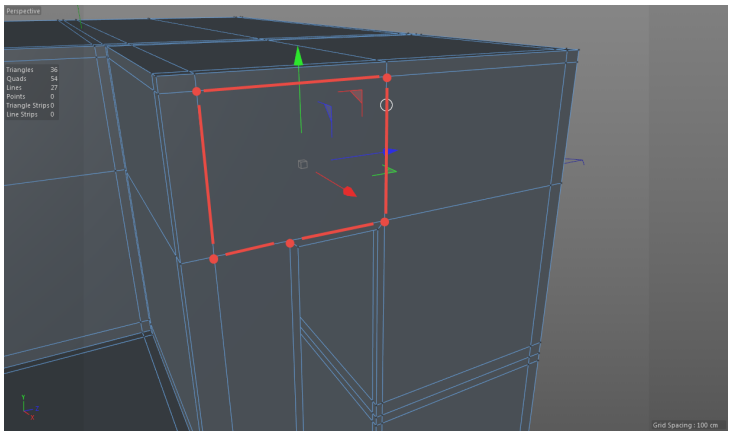
How to Turn N-Gons into Quads

First, in Cinema 4D go to the viewer **Option -> Configure -> HUD** tab. Activate **Total N-Gons**. If there are any in your scene, you'll see a HUD element on the top left of the viewer. If there are any, look for any polygons with more than four verts on your model.

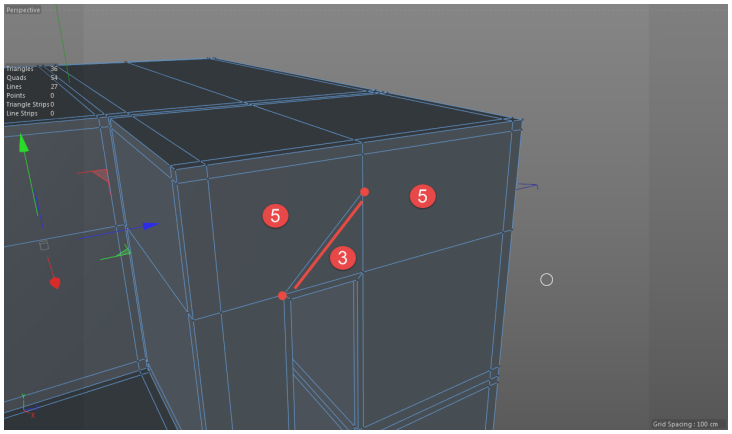
You'll need to break any N-Gons into quads. This isn't always easy.

1. First, try extending the fifth vert into it's own loop around the whole object.
2. If that's not possible, try converting the N-Gon into quads by cutting a triangle (see images overleaf) and cutting that into a quad. This will create a pole, which is okay if it's off a curve.
    a. You'll often need to create an additional loop.
3. If it's on a curve, you'll need to move the pole to a flat surface. This will take some cutting and dissolving of edges.

On the following page, we'll look at it in more detail at an NGon issue.

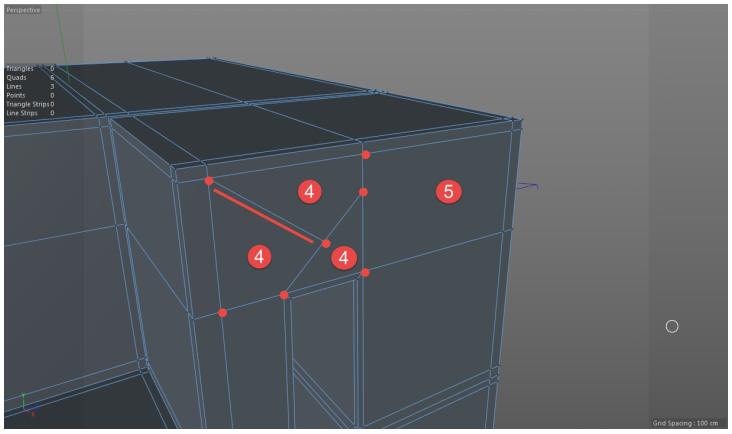


Here is the NGon.



First, we cut a new triangle.

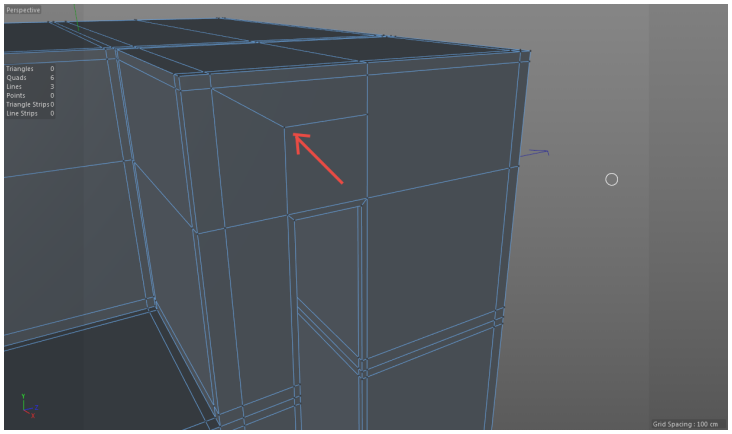
But that leaves us with two NGons (marked 5) and a triangle (marked 3).



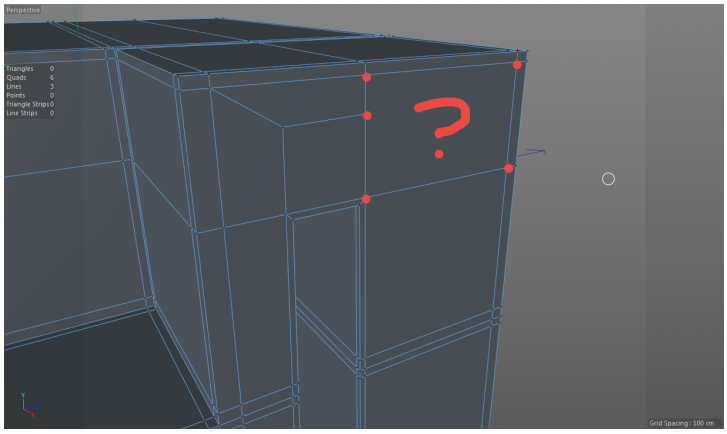
So we cut the triangle from the corner, leaving us with three quads (marked 4) and one NGon (marked 5). Let's call this the "cut a triangle to quad" technique.

At this stage, all we've done is move the problem.

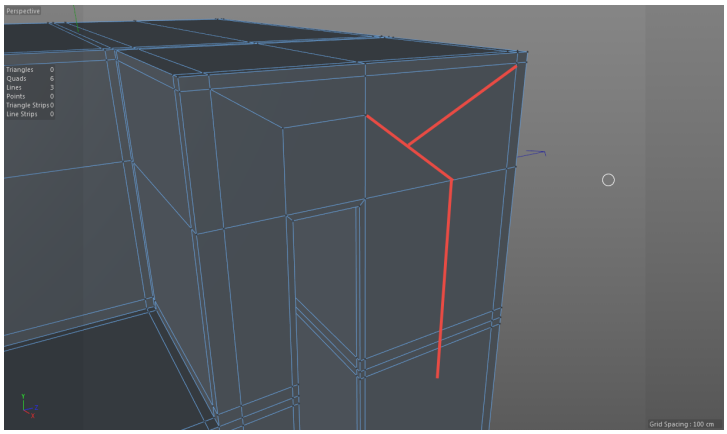
But we're not finished yet.



First, let's move that vert so that it looks more like a square than a triangle (we don't want any t-verts).

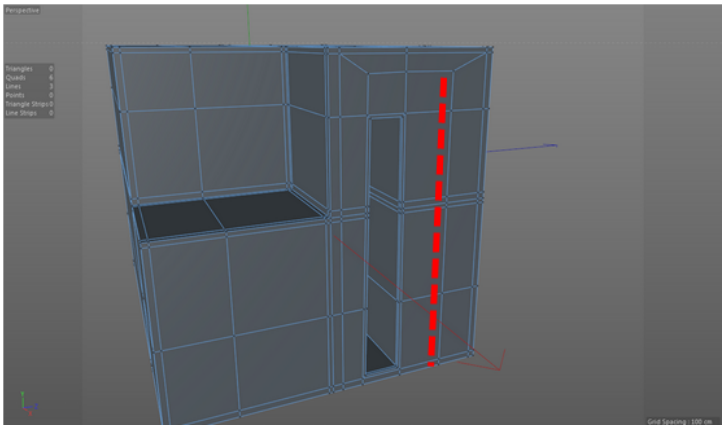


Here is that new NGon that we want to get rid of.

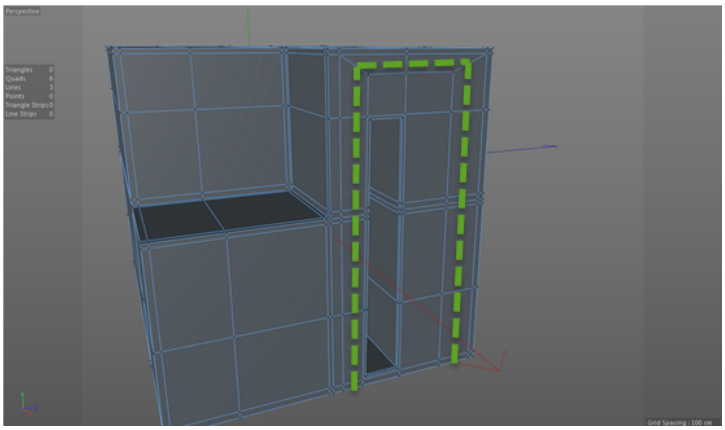


We do the same “cut a triangle to quad” technique on the new NGon (just like we did before) and shape the new quad into a square.

We also cut the bottom vert of that new quad all the way down to the base of the object.



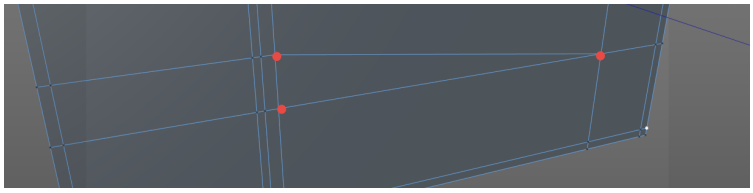
When you cut this line through all the polygons all the way to the base, you get rid of all the NGons and triangles.



This result in a loop. Which is great.

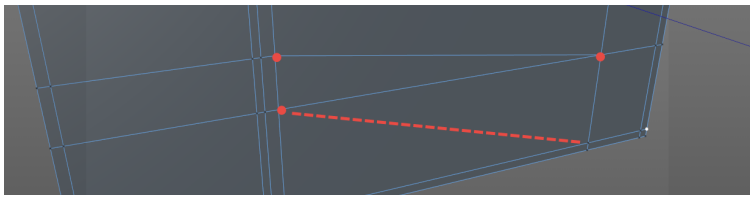
How to move triangles into hidden areas

Triangles are okay if they're hidden. Try to move triangles to hidden areas by cutting a

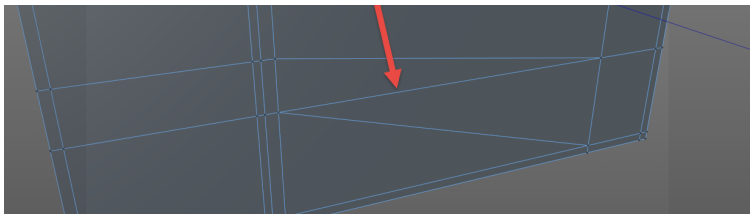


new triangle and then dissolving the old edge.

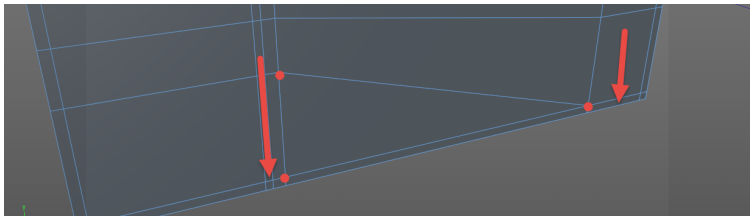
Here is a triangle.



If you cut a nearby quad you'll end up with three triangles.



See the three triangles here? You can grab the edge between two triangles and remove it.

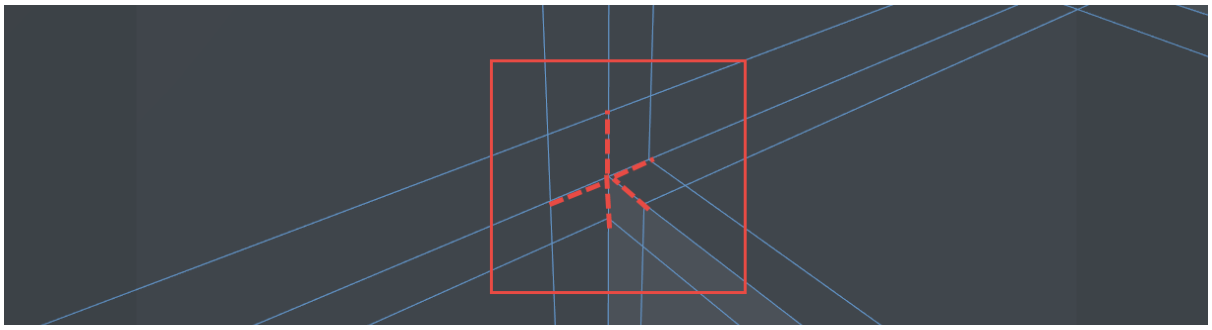


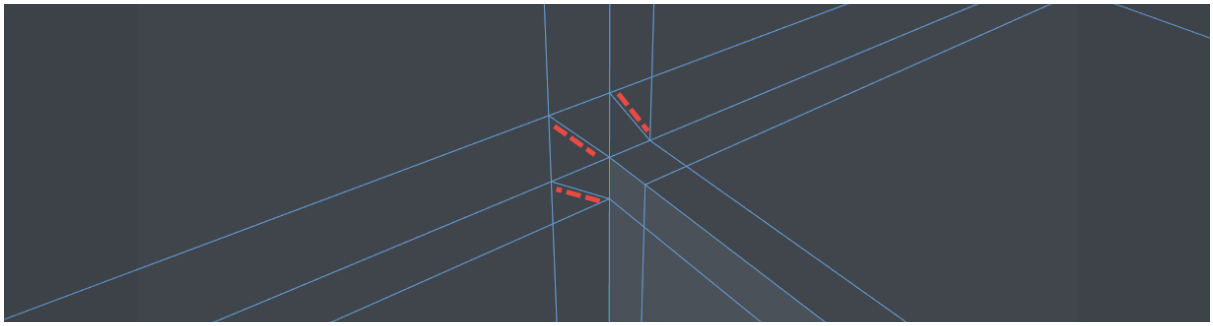
After you remove that edge, you're back to having a single triangle. See how it's lower down than it was? That's how you move a triangle.

How to Move Poles off Edges

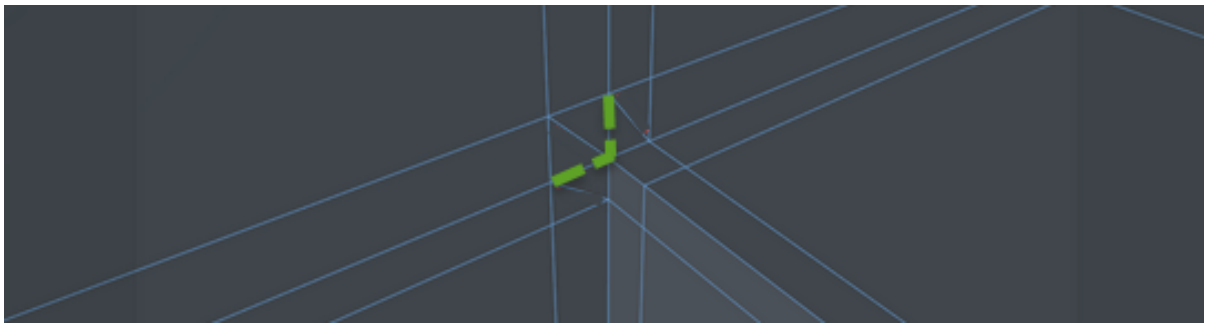
Move all your poles away from edges.

In the following sequence, there is a pole on a corner. This is pretty common. Let's move that pole elsewhere.

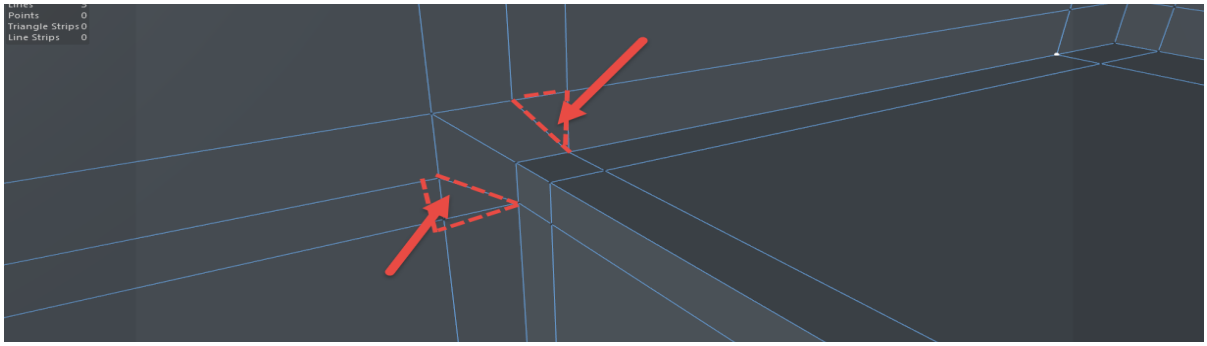




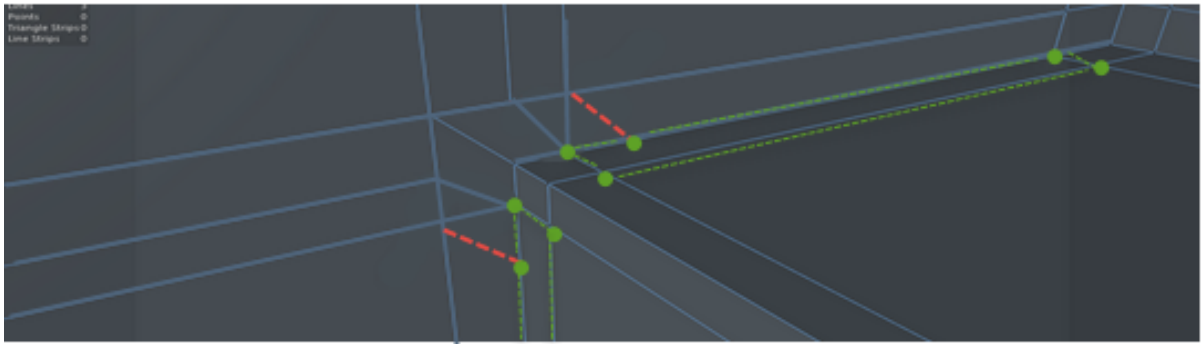
First we cut three new edges (above).



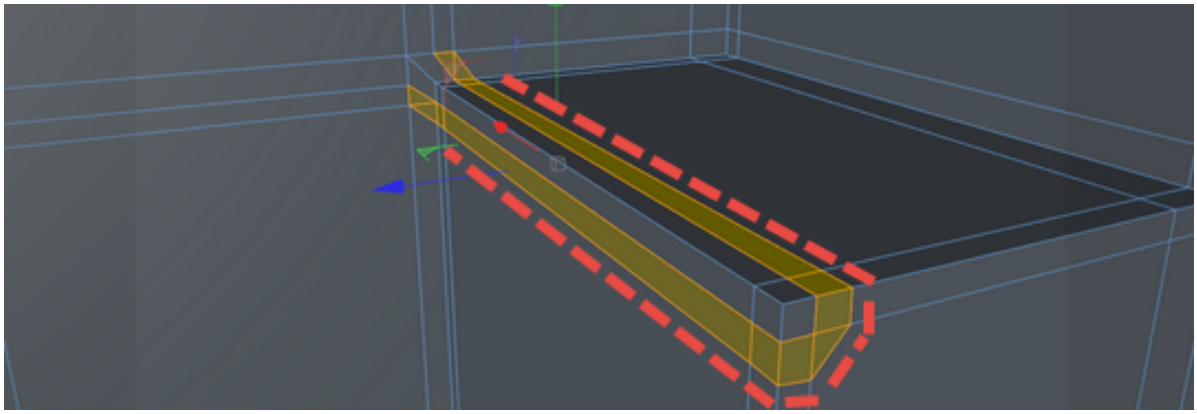
Then we remove two edges (above), which gives us a clean corner.



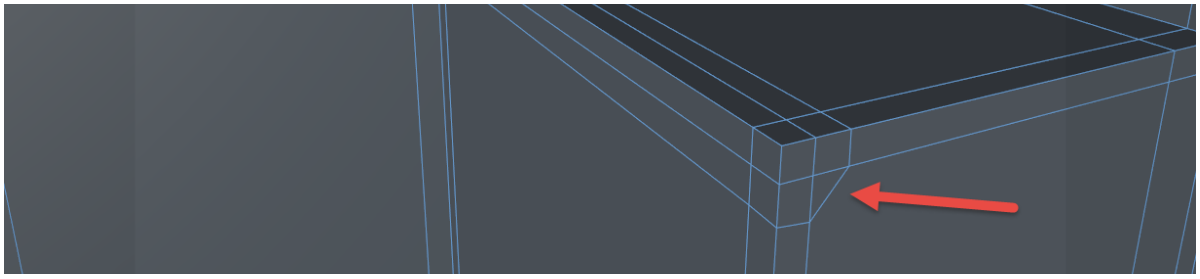
But that leaves two triangles which we'll need to remove.



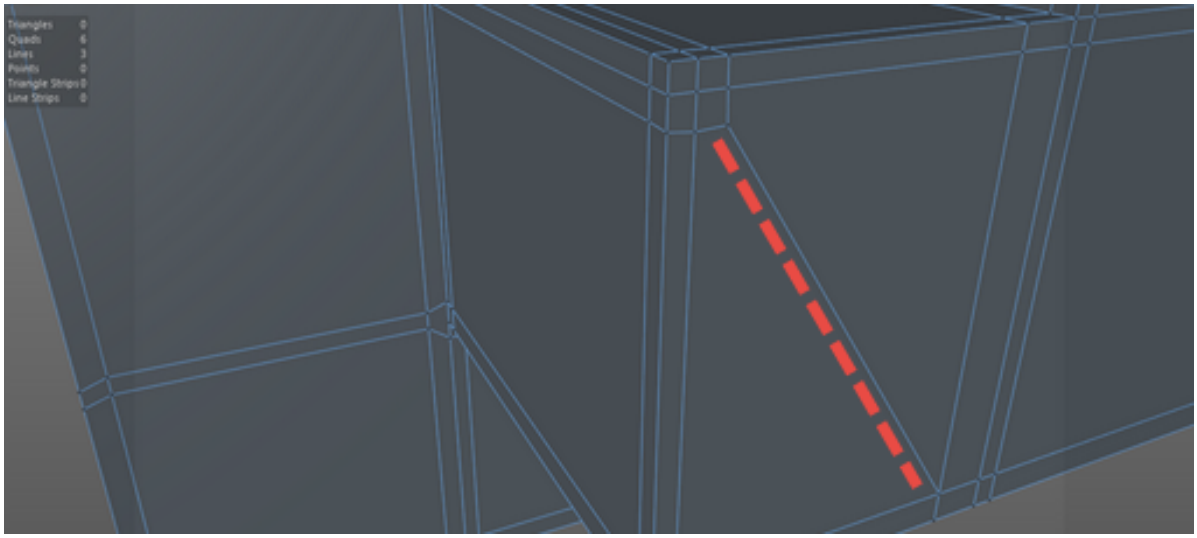
We can cut a new line down (see red lines) from those triangles to turn them into quads. This creates some crappy N-Gons (see green elements), of course.



But that's not an issue because we can extend those cuts. Above, we've cut the red line all around so the lines meet up to form a loop.



Now we have a rogue triangle in the loop. Let's get rid of that.



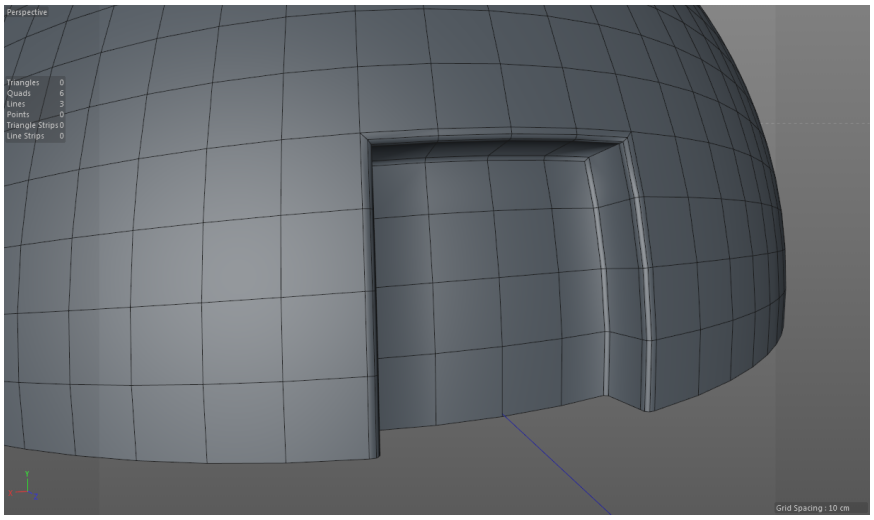
We use the usual technique for triangles … We cut the triangle into a quad and make an

edge into a nearby corner. Then pull the new vert out so it's not a t-vert.

How to deal with Poles on Curves

Always move your poles off curves.

If this is impossible (because you're modelling on a curved surface), consider adding more resolution before adding extrusions to the curve, so you can use the natural curve lines (instead of isolines) to add extrusion.

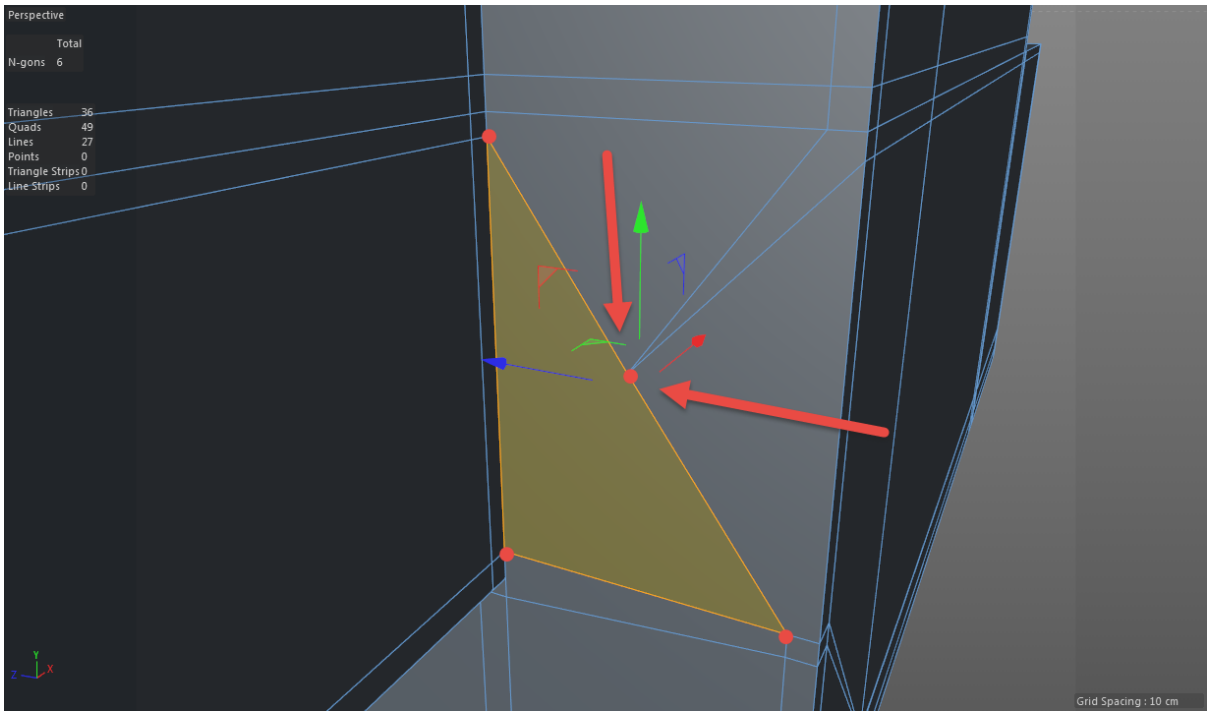


Basically, give yourself enough loops to model the curve accurately in the low poly mesh (don't rely on the subdivision to provide the curve).

Or, if the curves are shallow enough, you can spread your poles to the corner as though it were a flat surface.

Getting rid of T-Verts

Make sure your four vert polys look like squares, **not** triangles like the in the image below.



Chapter 4: UV & Texturing Workflows

First: Unwrap UVW in Cinema 4D

Unwrapping UVs should be thought of like you're peeling an orange and laying the skin out flat (using the fewest pieces possible). You've got to cut the skin, peel it back, and lay it flat.

Here's the workflow for Cinema 4D.

- **Select edge seams where you are making the "cuts"** (imagine how would have to cut it if you would have to unfold it flat).
    - *Always* cut seams where there are hard edges.
        - If you don't, you'll get bad normal maps later down the line.
    - Try to make other seams in areas normally out of sight.
        - Seams on soft edges can sometimes give you texturing problems … so keep them out of sight if possible.
- **BP UV Edit,** go to **UV Polygons**. **Select all**.
- Hit **Projection -> Frontal.**
- **Relax UV** with **Pin to Neighbours** and **Cut Selected Edges.**
- **Rotate** all the pieces (select a piece with ALT+click in Live Selection mode).
- Use a checker texture to view your UV spacing. If the numbers and letters looked flipped, use mirror U and mirror V to get them looking correct.
- **Optimal Mapping** with **Realign**, **Preserve Orientation**, **Equalize Island Size** and 2% Spacing (larger, like 3%, if you're creating a normal sculpt, see below).
- If there are bad sections, edit the seams and try again.
- Or use frontal projection from a specific view and apply a relax without cut selected edges (and maybe even Pin Border Points).

Second: Export an FBX mesh from

Cinema

Before you can texture your object in third party applications, you'll need to be able to export it. Here's how to export the low poly mesh from Cinema.

1. Make sure your object is in the correct **scale** (edit project settings and set the units to meters) and is sized appropriately.
2. **Isolate** the object.
3. Remove all textures apart from the colour map (**only tick the colour checkbox**). If you're using a colour texture, set your **diffuse colour to white**.
4. Make it **editable**.
5. **Export using latest FBX**.
    a. Tick **Normals**.
    b. Tick **Triangulate geometry**.
    c. Tick **Textures and Materials**.
    d. Tick **Embed Textures**.
6. Leave all other checkboxes unticked.
7. Hit OK.

Third: Bake Normals and AO from a High Poly Mesh (if you have one)

There are several methods for normal baking and ao baking, which we'll explore here.

If you don't have a high poly mesh, you can still follow these procedures. Just use your low-poly mesh in place of a high poly mesh (it isn't a great solution but it kind of works).

BodyPaint - Normal Map and AO Baking

You can add detail to your low poly mesh using BodyPaint in Cinema 4D. This method requires a sculpting modelling process.

1. Split the object into pieces where the mesh has a lot of complex detail.
2. Use Sculpting to add subdivisions. Sculpt using the available tools.
3. For example, use the mask mode and a flood inflate to add hard surface detail.
4. Turn on Ambient Occlusion as an Effect in the render settings. Set Maximum samples to 256 and maximum ray length to something like 500 cm.
5. Render the object to see how it looks. Change AO settings if required.
6. Use the Bake Sculpt Objects tool to bake out a normal maps and an ambient occlusion maps for the various splits.
    a. Choose an output file, use 16 pixel border, choose Normal and Ambient Occlusion in the Options tab, choose Tangent for the method in the Settings tab. Hit Bake.

Substance Painter - Normal Map and AO Baking

If you would prefer to model the high poly version (rather than sculpt) and bake it out, you can use a program like Substance Painter or xNormal.

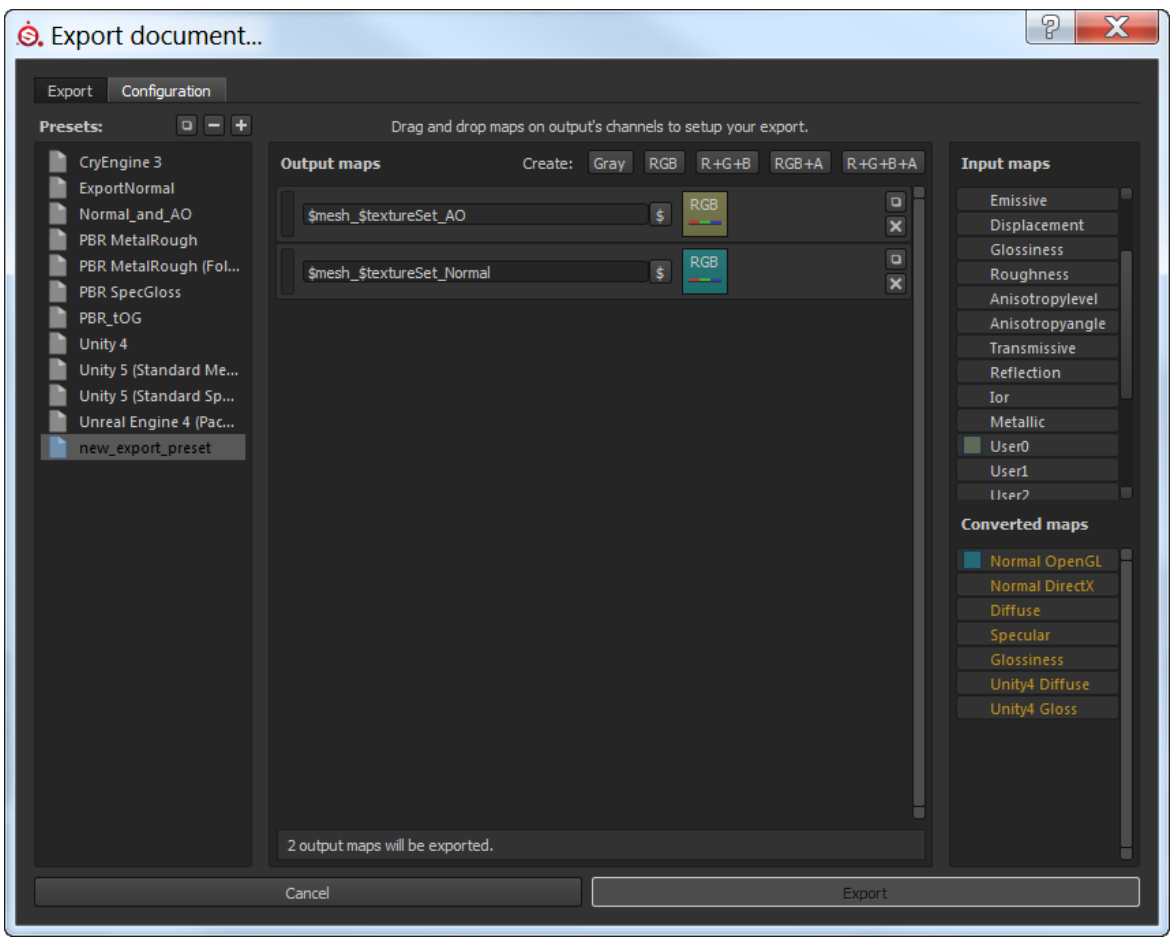
We're going to talk about Substance Painter here.

This process requires a third mesh, called a **cage**—this is a copy of the low poly object, with all its points inflated (pushed out along their normals) so that it completely surrounds the high poly mesh.

1. Model and export the low poly mesh.
    a. Add the suffix "_low" to your low poly mesh, e.g., "ball_low".
    b. Remove all materials except for the colour (sometimes called diffuse) map.

    c. Export the low poly mesh as an FBX.
        i. Tick Normals.
        ii. Tick Triangulate geometry.
        iii. Tick Textures and Materials.
        iv. Tick Embed Textures.
        v. Leave all other checkboxes unticked.
        vi. Hit OK.
2. **Model the high poly mesh**. Add sculpting if desired for high frequency details.
    a. Add the suffix "_high" to your low poly mesh, e.g., "ball_high".
    b. Export the high poly mesh as an FBX (see 1.c).
3. **Model the cage object**.
    a. Duplicate the low poly mesh and give it the suffix "_cage".
    b. Inflate (or "push" in 3DS Max) the cage object vertices so that it completely surrounds the high poly mesh.
    c. Export the cage mesh as an FBX (see 1.c)..
4. Start a new project in **Substance Painter**. Use the Low Poly mesh and 2048 document resolution.
5. Use **Bake Textures**.
    a. Tick **Normal.**
    b. Tick **Ambient Occlusion**.
    c. Set your output size to 2048.
    d. Choose a **dilation width** of around 1.
    e. Select the **High Definition Mesh** by hitting the new icon and pointing to your high poly mesh FBX on disk.
    f. Ticke **Use Cage**.
    g. Select the **cage mesh** by hitting the new icon in the Cage File field and pointing to your cage mesh FBX on disk.
    h. Change **Match** to By Mesh Name.
    i. Put in some **Antialiasing** (4x4 is fine).
    j. Hit **Bake Textures** and view your normal and AO within Substance Painter.
6. To **export the maps** for use in tOG …
    a. In the TextureSet settings, click the + button beside Channels and add a new user0 Channel.
    b. Change the format to L8.
    c. Create a fill layer.
    d. Turn off all channels in the fill except for the User0 channel.
    e. In the Textures tab in the shelf, find the AO map that was automatically generated. Drag this into the new fill layer's user0 channel.
    f. Use Export all Channels from the File menu. Go to the configuration tab.

g. Create a new configuration by hitting the + button in the configuration tab (see the image below).
    i. It will be called “new_export_Preset” in the Preset list on the left. Rename this now to “tOG Normal AO Export”.
    ii. Create an RGB output (click RGB) and drag the user0 (on the right) into it. Select RGB Channels..
    iii. Rename it from “RGB” to $mesh_$textureSet_AO.
    iv. Add a new RGB output (click RGB) and drag the Normal OpenGL (on the right) into it. Select RGB Channels.
    v. Rename it from “RGB” to $mesh_$textureSet_Normal.
h. Go back into the Export tab and change the output path to a suitable location.
i. Also change the output format to png (16 bits max).
j. Change the Config to your new “tOG Normal AO Export”.
k. Set the output size to 2048 x 2048 (or higher, if desired). Hit Export.



Substance Painter - Baking Multiple Objects at Once

If you have multiple objects that need to affect one another when generating AO, you will need to perform a few extra steps.

1. Build all your meshes and add the suffixes for low poly, high poly and your cages.
2. Make sure each object has its own, unique material name.
3. Export your low polys as a single FBX.
4. Export you high polys as a single FBX.
5. **Export your cages as individual FBXs.** This is important.
6. Use Substance Painter as before, but for each object you'll need to select its appropriate cage file—going to Bake textures and choosing the right cage file before you hit Bake Textures for each textureSet.

Otherwise it works just the same.

Fourth (optional): Add Normal Detail with a Height Map

Adding Detail in xNormal

To create a normal map from a height map without using Substance Painter, load up xNormal and Photoshop.

- In **Photoshop**, start with the AO map and add some surface detail in black and white as if you were creating a bump map.
- In **xNormal**, go to **Tools -> Height to Norma**l.
- **Right click -> Browse Height Map** in the left pane.
- **Right click -> Generate** in the right pane.
- **Right click -> Copy** the right pane.
- In **Photoshop**, paste it into your normal map as a new layer and save it out as a png.

Adding AO in xNormal

It is also possible to use xNormal to convert your height map into an ambient occlusion map using one of the tools or to bake an ambient occlusion map from a high poly mesh. However, that is beyond the scope of this document. For details on using xNormal, please

Fifth: Texture & Prepare Images for Import

Photoshop Texturing: Diffuse & Specular

Once you've got your normal map and your AO, go into Photoshop and start painting.

- Use the AO as a guide to paint blocks of colour.
- Also consider baking out the UV mask as a guide (using the Bake Texture dialog in Cinema 4D, for example).
- Paint your diffuse map. This technique relies on a sensible UV unwrap and some good Photoshop texturing skills.
- When you're finished with the diffuse map, it's time to create the specularity map.
- Consider using your AO as the basis for the specularity map. Keep it dark for non reflective materials and light for very reflective materials. Consider adding in high frequency surface detail here.

Photoshop Texturing: PBR Export

Normally this process is handled by your PBR image creation software (Quixel Suite or Substance Painter). However, for artists without access to that software, here is how to prepare the images for tOG:

1. **Name your images nicely before import.** Be clear. Make sure they're unique—try using a simple naming format like "Graphic_Mesh_Map.ext", e.g., "Formation_Plinth_Normal.png."
2. **If you don't have a normal map**, create a PNG that's 128,128,255 and call it "Generic_Normal_Blank.png".
3. **Combine your ambient occlusion (ao), roughness and metallic maps together**:
    a. **Create a new image**. Call it "Graphic_Mesh_AOMetalRough", e.g., "Formation_Plinth_AOMetalRough".
    b. Go to the **Channels** window.
    c. Select the "**Red**" channel.
    d. Open your **AO image**, select all, copy it and **paste it into the Red channel** of

your new image.
   e. Select the “**Green**” channel.
   f. Open your **Metallic** image, select all, copy it and **paste it into the Green channel** of your new image.
   g. Select the “**Blue**” channel.
   h. Open your **Roughness** image, select all, copy it and **paste it into the Blue channel** of your new image.
   i. Select the RGB channel.
   j. Save the image.

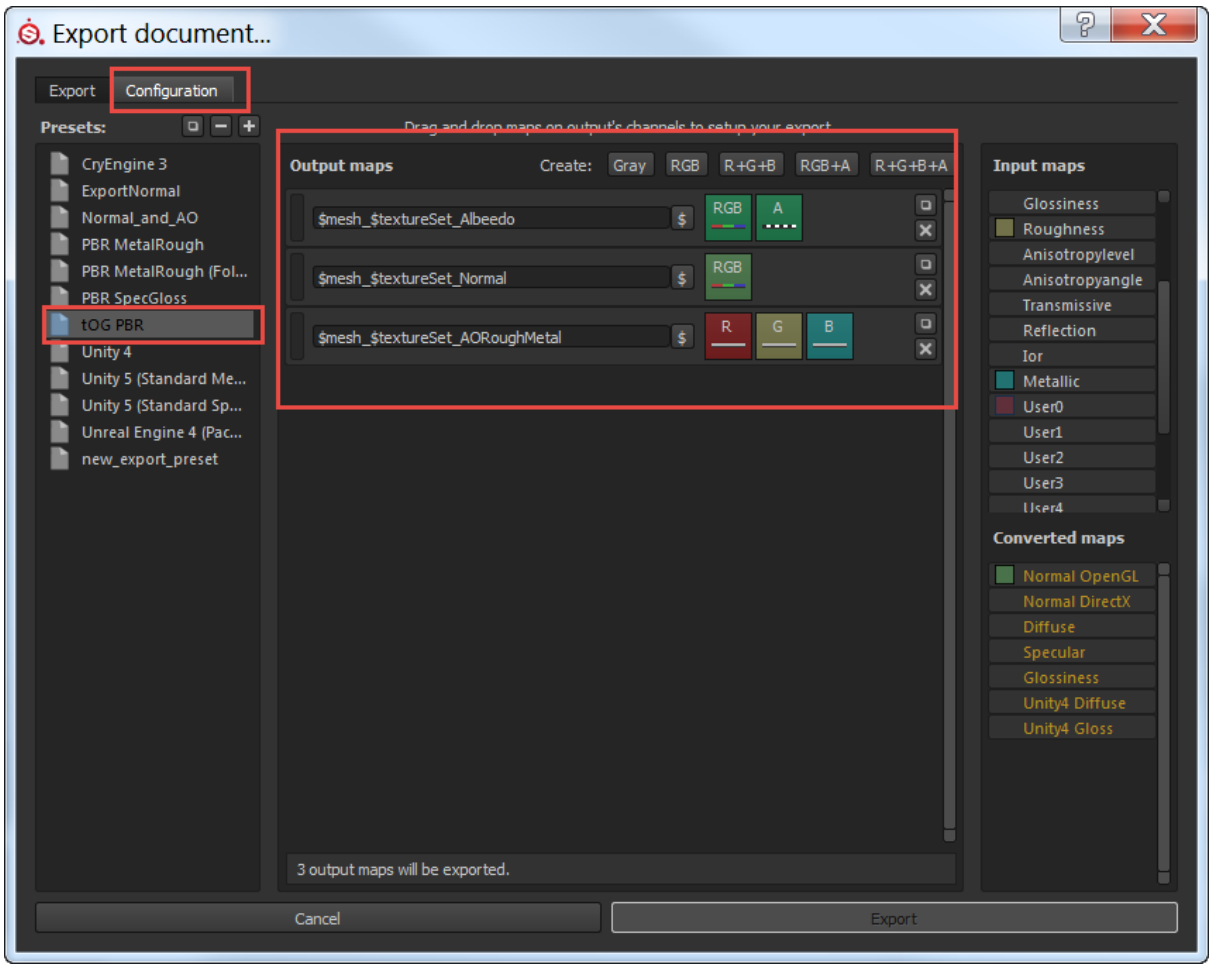
Substance Painter Texturing: PBR Export

If you are using Substance Painter, you can export the images as follows:

1. In the TextureSet settings, click the + button beside Channels and add a new user0 Channel.
2. Change the format to L8.
3. Create a fill layer.
4. Turn off all channels in the fill except for the User0 channel.
5. In the Textures tab in the shelf, find the AO map. Drag this into the new fill layer’s user0 channel.
6. Use Export all Channels from the File menu. Go to the configuration tab.
7. Create a new configuration by hitting the + button in the configuration tab (see the image below).
   a. It will be called “new_export_Preset” in the Preset list on the left. Rename this now to “tOG PBR”.
   b. Create an RGBA output (click RGB+A) and drag the BaseColor (on the right) into it the RGB. Select RGB Channels.
   c. Drag the BaseColor into the A and select A channel.
   d. Rename it from “RGB Alpha” to $mesh_$textureSet_Albeedo.
   e. Now add an RGB output (click RGB) and drag the Normal OpenGL (on the right) into it the RGB. Select RGB Channels.
   f. Rename it from “RGB” to $mesh_$textureSet_Normal.
   g. Lastly add an RGB output (click R+G+B) and drag the User0 (on the right) into it the R. Select Gray Channel.

h. Drag the Roughness (on the right) into it the G. Select Gray Channel.
i. Drag the Metallic (on the right) into it the B. Select Gray Channel.
j. Rename it from “R+G+B” to $mesh_$textureSet_AOMetalRough.

8. Go back into the Export tab and change the output path to a suitable location.
9. Also change the output format to png (16 bits max).
10. Change the Config to your new “tOG PBR”.
11. Set the output size to 2048 x 2048 (or higher, if desired).
12. Hit Export.



Chapter 5: Environment Maps

8bit Environment Map (for FR)

To generate a single exposure, real world spherical environment map for the Fresnel Reflection (FR) sader, we recommend using the Ricoh Theta.

- Use the device to capture a spherical map.
- **Resize** the image to an aspect ratio of 2:1 (e.g., 2048px x 1024 px).
- Use **curves** in Photoshop to brighten the highlights and *severely* darken the shadows and midtones.
- If you're going to use the Fresnel Reflection shader, consider **blurring** the environment (you may also want to create a blurred environment if you're creating a skybox). In Photoshop.
    - **Unlock** and **duplicate** the background.
    - **Offset** the new layer by 50% horizontally.
    - Convert the new layer to a **smart object**.
    - Convert the old layer to a **smart object**.
    - Apply a **gaussian blur** to both layers (around 20 pixels).
    - Select the top layer and **offset** it back 50% horizontally.
    - You'll see a bad seam. Drag a **selection** around it and feather it by 60 pixels or so.
    - Invert the selection a and apply it as a **mask** on the top layer.
    - You can play with the gaussian blur strength now, since both your layers are smart objects.
    - Flatten the image when you're done.
- Consider adding detail back into the blurred environment with an overlay.

Virtual 8bit Environment Map

If you are generating your environment map for a virtual set, you will need to generate the map using your 3D software package. This is often achieved by:

- Creating a spherical camera and rendering your scene.
    - Render an image with a 2:1 ratio, e.g., 2048px x 1024px.
- Creating a sphere with 100% reflection and “baking” the reflection to a file.
    - Bake the texture to an image with a 2:1 ratio, e.g., 2048px x 1024px.

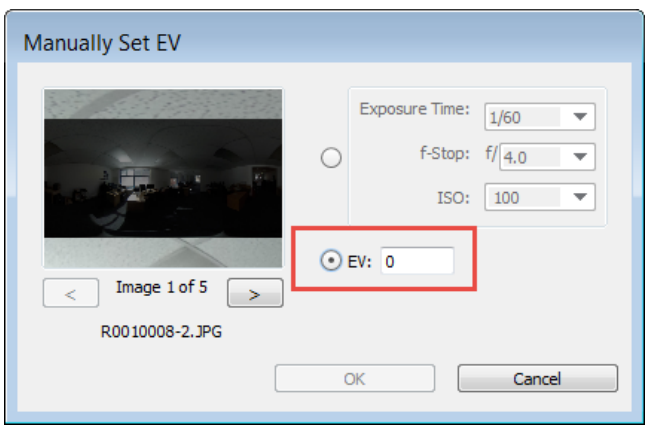
Teaching how to create a spherical environment map specifically from a 3D package is beyond the scope of this tutorial.

32bit Environment Map (for PBR)

This only applies to tOG 4.6.2 or later. If you are using an earlier version, you must use the 8bit method outlined above as HDR images are not fully supported until 4.6.2.

To generate a multiple exposure, real world spherical environment map for PBR, we recommend using the Ricoh Theta. You'll need to create a HDR from 3 exposures.

1. First take three exposures (-2, 0, +2) of the environment with a 360 degree camera.
2. Next, take the -2 and +2 exposures and use Photoshop's **Image -> Adjustments -> Exposure** to create a -4 and a +4 exposure.
3. Within Photoshop, go to **File -> Automate -> Merge to HDR Pro** …
4. Browse for your multiple exposures (including your new -4 / +4 ones) and hit OK.
5. In the resulting dialog, change to **exposure value mode (EV)**.



1. Set the EV of each image according to the exposure value of the shot (cycle through exposures using the left and right arrows).
2. Hit OK.
3. Make sure the **Mode** is set to **32 bit**.
4. Remove the tick from "Complete Toning in Adobe Camera Raw."
5. Hit **OK**. Save the image as a **Radiance file (hdr)**.

Virtual HDR Environment Map

Generating a 32bit VR environment map is much the same as an 8bit (see *Virtual 8bit Environment Map* above for suggestions). However, when you generate the image, you will need to output a Radiance file (*.hdr).

Cinema 4D - Rendering Geometry into the HDR Image

If you need to mix your geometry with a real world environment (when implementing AR), you can generate your HDR environment map within Cinema 4D. Here is a workflow for bringing virtual elements into a HDR:

http://lesterbanks.com/2012/09/create-hdris-from-your-own-cinema-4d-scenes/

1. Add HDR image based lighting to your scene (add a Sky with a Luminance material with your HDR image).
2. Create a sphere (which acts as as the 360 degree camera).
3. Create a fully reflective material and assign it to the sphere.
4. Add a Compositing tag to the sphere and uncheck cast shadows.
5. Add a Bake Texture tag to the sphere. Set it to only use Reflection in the Options tab.
    a. In the tag settings, change Height to 2048 and the Width to 1024.
    b. Change the format to Radiance (HDR).
    c. Choose a filename for the output.
    d. Go to options, click Bake.

Now you have a new HDR map which merges your scene geometry into the original HDR.

Chapter 6: Meshes and Animation

Mesh Workflow

Export an FBX mesh from Cinema

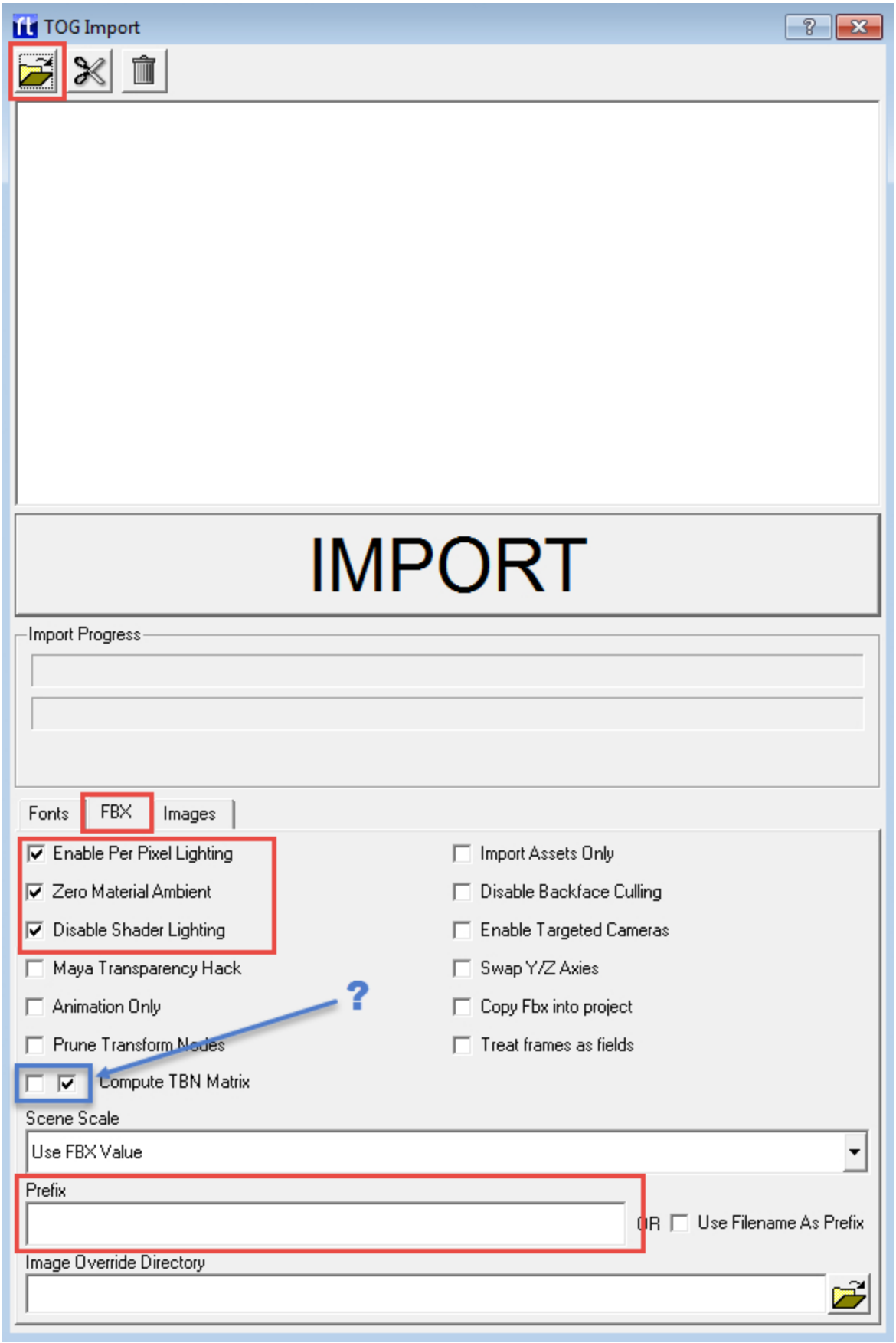
Before you can texture your object in third party applications, you'll need to be able to export it. Here's how to export the low poly mesh from Cinema.

1. Make sure your object is in the correct **scale** (edit project settings and set the units to meters) and is sized appropriately. **Isolate the object.**
2. **Rename the object** to something appropriate. Make sure this name is unique (consider naming it according to the destination graphic, "Graphic_Mesh." For example, when naming the mesh that will be used in a leaderboard graphic, call it "Leaderboard_TabelRow."
    a. If you do not name your mesh uniquely, you will overwrite other meshes during import!
3. **Remove all textures** apart from the colour map.
4. **Rename the material** to something short and descriptive e.g., "Colour" or something to that effect.
5. Make it **editable**.
6. **Export** using FBX 2014.
    a. Tick **Normals**.
    b. Tick **Textures and Materials**.
    c. Tick **Embed Textures**.
    d. Leave all other checkboxes unticked.
    e. Hit OK.

Import an FBX into tOG Edit

To import an FBX into tOG …

1. Go to **Graphic -> Import**.
2. Click on the **yellow file browser**, browse to your FBX and click Open.
3. Go to the **FBX tab** (see image below).
    a. Tick **Enable Per Pixel Lighting**.
    b. Tick **Zero Material Ambient**.
    c. Tick **Disable Shader Lighting**.
4. If you used Cinema 4D to export your meshes, tick the Compute TBN Matrix option.
5. Otherwise, leave the Compute TBN Matrix option blank.
6. Consider adding a **prefix** if it will help you to manage your object within tOG (seee the Prefix field in the image below).
7. Hit **Import**.


TOG Import
IMPORT
Import Progress
Fonts
FBX
Images
Enable Per Pixel Lighting
Zero Material Ambient
Disable Shader Lighting
Maya Transparency Hack
Animation Only
Prune Transform Nodes
Compute TBN Matrix
?
Import Assets Only
Disable Backface Culling
Enable Targeted Cameras
Swap Y/Z Axies
Copy Fbx into project
Treat frames as fields
Scene Scale
Use FBX Value
Prefix
OR
Use Filename As Prefix
Image Override Directory

Troubleshooting: The mesh looks black

Bad Scale?

First make sure the object is the right scale.

1. Enable “Background Checker  Board” in the viewer options.
2. If the viewer is still black after you click this button, your mesh is not being lit correctly (see below).
3. However, if the viewer shows a checkerboard and no object, the scale of your object could be wrong.
    a. Try applying a uniform scale to your object to scale it down or scale it up until you can see the object.
    b. Alternatively, re-export your object with a 1m per unit scale. Make sure your object is in the correct scale (edit project settings and set the units to meters) and is sized appropriately.

Zero Diffuse?

Check your shader’s meterial tab and see if the diffuse colour is set to 0.0, 0.0, 0.0. If it is, try setting it to 1.0,1.0,1.0.

Lighting Enabled?

First, make sure lighting is disabled in the shader. If you need lighting from the scene, you can enable the shader lighting but you’ll also need to enable a light source.

1. Go to the ShaderNode tab in the shader.
2. Go to the Lighting rollout.
3. Tick a light.

If this doesn’t fix it, it could be something to do with the imported mesh.

Missing Tangents and Bitangents?

If you are using an imported mesh (from an FBX) and you're using an advanced shader (like PBR or FR), it's possible that the mesh is missing some important information called tangents and bitangents (or binormals). These are normally defined during export but sometimes the exporting software doesn't export them properly. However, there is a way to compute the tangents and bitangents within tOG Edit.

1. Click on the geometry in the scene graph.
2. Click **Compute TBN Matrix**.
3. Click on the **Save icon** in the Geometry pane.
4. Enter the name of the mesh with the suffix "_TBN", e.g., if the mesh was called Cube, save it as "Cube_TBN".
5. **Save the project and close it.**
6. Reopen the project and the graphic.
7. If this hasn't fixed your problem, please email support@rtsw.co.uk with details of your problem (a screenshot, the project files and details of your version of tOG).

Object Animation Workflow

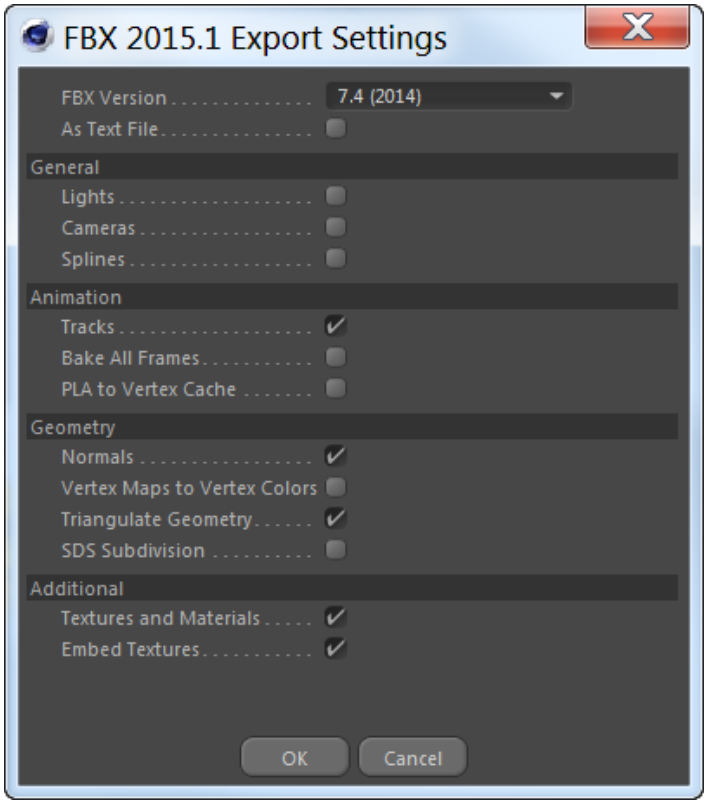
There are two ways to bake out animation from Cinema 4D. The first uses Nitrobake, a third party plugin.

Baking Animation with NitroBake

This workflow assumes that each object you are baking will have it's own UV coordinates (as opposed to lots of objects sharing a single set of UVs). Please note that because the objects are animating, you will not be able to bake out ambient occlusion for these sequences.

1. Build your animation sequence using the MoGraph tools.
2. Select the parent node of the animated objects.
3. Open the NitroBake plugin.
4. Click NitroBake.
5. The animation will now be baked to a new object.
6. Get rid of everything else in the scene and save as a new Cinema 4D file. Make sure your object is in the correct scale (edit project settings and set the units to meters) and is sized appropriately.
7. **Rename the nodes of the object** to something appropriate. Make sure this name is unique (consider naming it according to the destination graphic, "Graphic_Mesh." For example, when naming the mesh that will be used in a leaderboard graphic, call it "Leaderboard_TabelRow."
    a. If you do not name your mesh uniquely, you will overwrite other meshes during import!
8. **Remove all textures** apart from the colour map.
9. **Rename the material** to a unique name. Like the mesh name, make it unique and obvious, e.g., "Leaderboard_TableRow_Dark" or something to that effect.
    a. If you do not name your material uniquely, you will overwrite other materials during import!
10. **Export** using FBX 2014.
    a. Tick **Tracks.**
    b. Tick **Normals**.
    c. Tick **Triangulate geometry**.

d. Tick **Textures and Materials**.
e. Tick **Embed Textures**.
f. Leave all other checkboxes unticked.
g. Hit OK.

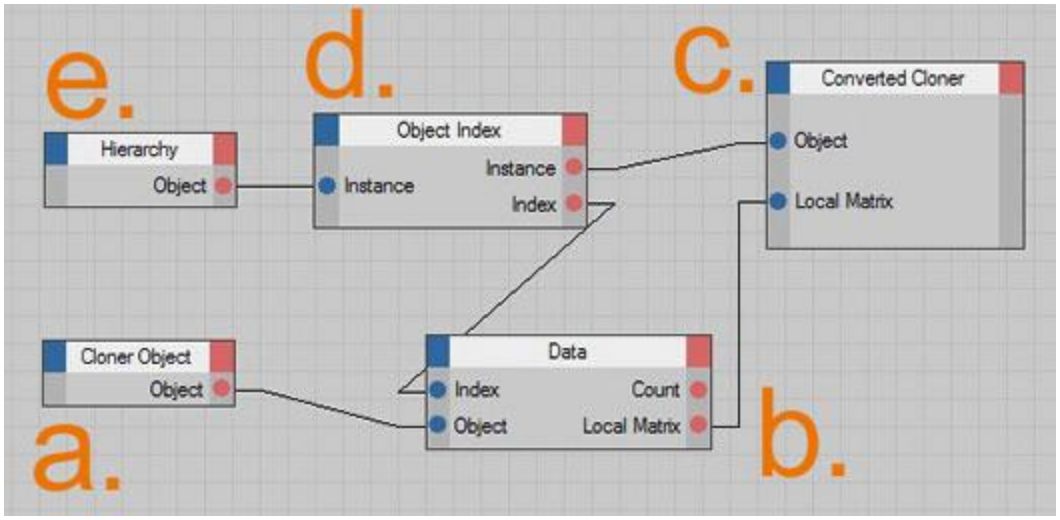


Bake Animation using Xpresso (The Ramsay Technique)

To bake out mograph animation to keyframes

http://www.jag4d.com/bake_mograph_objects_to_keyframes.html

1. **Select** the Cloner/Fracture Object in object manager.
2. **Duplicate** your Cloner / Fracture Object.
3. **Hit "C"** to convert the Cloner to single objects, and **rename** it to 'Converted Cloner' (for example)
4. Put a **XPresso Tag** (Cinema 4D Tags>XPresso) on the Converted Cloner which attaches every "physical" copy of your objects to its cloned counterpart. *N.B. This expression must be set to 'Generator' to evaluate precisely (under 'priority' in the xpresso tag).*
5. Open the editor by **double clicking the XPresso** tag,
    a. **Drag** the Cloner Object into the editor, add object **output** to Cloner Object node (click on the red rectangle>select object output)
    b. Create a **Data** node (right click in Xpresso editor>New Node>Motion Graphics>Motion Objects)
    c. **Drag** the Converted Cloner Object into the editor, add **object** and **local Matrix** inputs to Converted Cloner Object node (click on the blue rectangle>select object input and local matrix input)
    d. Create a **Object Index node** (right click in Xpresso editor>New Node>XPresso>General)
    e. Create a **Hierarchy node** (right click in Xpresso editor>New Node>XPresso>Iterator), add object output to Hierarchy node (click on the red rectangle>select object output)
6. Now **Connect** them all (see image below).



7. Be sure you have the converted Cloner **selected** in Object Manager (the Null that contains all your "physical" copies).
8. Activate the **axis tool**, go into **scale mode** (use scale tool) and click the **Cappucino** (Character > Manager > Cappucino)
    a. Select **Rewind Time**, **Position**, **Rotation** and **Hierarchy**.
    b. Hit **Start Realtime.**
9. **Click and hold the LMB** anywhere in your editor view until the end of the animation (do not drag the mouse during this time!).

Export Animation with Shared UVs

When you want to use many elements to combine together to make up a single image, for example when the front side of a wall of cubes shows one image and back side shows a different image.

1. **Add your material to the top object** (parent of the baked elements). Set **projection of the texture tag** to camera mapping. Create front **camera**, and drag this camera into camera box in the texture tag.
2. **Delete any UV tags** on the children objects. Now right click the parent object and use **Copy tag to Children**. You can **delete the texture tag** on the parent now.
3. Duplicate the parent, rename them 'Front' and 'Back' (for example). Select 'Front'. Move to where you want to make the first projection in the timeline. Select all the texture tags and right click- select **Generate UVW coordinates**.
4. Select 'Back'. Move to where you want to make the second projection in the timeline. Select all the texture tags and right click- select **Generate UVW coordinates**.
5. Delete all the back faces from 'Front' and the front faces from 'Back'.

6. As per Baking Animation with NitroBake, get your **naming** right!
7. **Remove all textures** apart from the colour map. Make sure texture tag is the last tag on each object.
8. Make sure your object is in the correct scale (edit project settings and set the units to meters) and is sized appropriately.
9. **Export** using FBX 2014.
    a. Tick **Tracks**, **Normals**, **Triangulate geometry**, **Textures and Materials** and **Embed Textures**.
    b. Leave all other checkboxes unticked. Hit OK.

Import Object Animation

Open the graphic that you want to import into (or create a new graphic). Follow the Import FBX into tOG Edit workflow above. This will create a new block of animation in your method. If you run into any mesh visibility problems, check the Troubleshooting: The Mesh Looks Black section above.

Object Deformation Workflow

If you need to deform an object (bake a cloth simulation, melt it, change it's shape, form into another object, bulge it, etc.), you use NitroBake and RiptidePro to bake out an FBX sequence.

Export a Mesh Sequence from Cinema 4D

1. Make sure your object is in the correct scale (edit project settings and set the units to meters) and is sized appropriately.
2. **Isolate the object.**
3. **Rename the object** to something appropriate. Make sure this name is unique (consider naming it according to the destination graphic, "Graphic_Mesh." For example, when naming the mesh that will be used in a leaderboard graphic, call it "Leaderboard_TabelRow."
    a. If you do not name your mesh uniquely, you will overwrite other meshes during import!
4. **Remove all textures** apart from the colour map.
5. **Rename the material** to a unique name. Like the mesh name, make it unique and obvious, e.g., "Leaderboard_TableRow_Dark" or something to that effect.
    a. If you do not name your material uniquely, you will overwrite other materials during import!
6. Use **NitroBake** to bake animation as a Point Animation. Isolate this baked animation (removing all other instances).
7. Use **Riptide Pro** to **export** as an OBJ sequence.
    a. Files:
        i. Animation Sequence
    b. Filter:
        i. Export Mesh Names

        ii. Export UV Coords (not compressed)
        iii. Export Normals (not compressed)
        iv. Flip Z axis
        v. Flip UV vertically
8. Use Riptide Pro to **import** the OBJ sequence back into C4D.
    a. Turn off Import NGons
    b. Turn on Import Normals
    c. Turn off Center Mesh Axis
9. **Order meshes by name** using the OrderByName script.
10. **Export** as an FBX.

Importing a Mesh Sequence

Import the FBX into your graphic (make sure to select Compute TBN if you exported the FBX from Cinema 4D) as per the Import FBX into tOG Edit section above.

Add a **Select node** above all the objects.

Use a **ramp animator** to animate the sequence:

> Add a ramp animator to the select node frame parameter.
>
> Set the start of the animator to frame 0.
>
> Set the end of the animator to the last frame (you can find this out by editing the select node's frame directly and finding the last frame before the mesh disappears).
>
> Set the length of the animator to the duration of the animation.

If the animation seems to be running backwards, the mesh ordering might be wrong. Try flipping the animator (or change the order manually in the scene graph).

If you run into any mesh visibility problems, check the Troubleshooting: The Mesh Looks Black section above.

Chapter 7: Complete Shader Workflow

Overview

This chapter is intended for VR / AR artists. These tutorials will help artists to generate relevant assets for use with one of the advanced FX shaders and instruct artists how to apply those shaders within tOG Edit.

It is intended for artists who are already familiar with tOG Edit. For more information about any of the topics covered here, please email support@rtsw.co.uk.

Shader Requirements

Fresnel Reflection (FR) Requirements

The fresnel reflection (FR) shader is a fast reflection shader that approximates reflections on an object loosely based on real world physics. To implement this shader you will need:

- The fx file: FresnelShaderSphere.fx (this should be placed in the GMData/CGPrograms/ folder of your project).
- An 8bit (or 16bit) per channel spherical environment map.
- A UV unwrapped 3D object with the following maps:
    - Diffuse.
    - Specularity.
    - Tangent Space Normal.

Physically Based Rendering (PBR) Requirements

The physically based rendering (PBR) shader is a robust shader that mimics physical lighting based on material properties. PBR requires an extensive set of maps. These are normally generated by third applications such as Quixel Suite or Substance Painter. **It is beyond the scope of this document to teach how to create PBR maps**.

However, once your PBR shader artists have created the appropriate maps, you can implement them within tOG Edit. To implement this shader you will need:

- The fx file: PBR_Spherical.fx (this should be placed in the GMData/CGPrograms/ folder of your project).
- A 32bit per channel, HDR, spherical environment map (.hdr file extension).
- A UV unwrapped 3D object with the following maps:
    - Albeedo.
    - Combined ambient occlusion (ao), roughness and metallic map. This is explained in more detail below.
    - Tangent Space Normal.

Applying FR in tOG

Time to apply a Fresnel Reflection shader.

1. **Name your images nicely before import.** Be clear. Make sure they're unique—try using a simple naming format like "graphic_mesh_map.ext", e.g., "Formation_Plinth_Normal.png."
    a. If you don't have a specular map for some reason, create a png that's pure white and call it "Generic_White.png".
    b. If you don't have a normal map, create a PNG that's 128,128,255 and call it "Generic_Normal_Blank.png".
2. **Import** your diffuse, specular and normal maps.
3. **Import** your environment map too!
4. Add an **Effect** node below your object.
5. Choose the **FresnelShader** from the dropdown within the Effect node.
6. In the shader for the object, go to the **ShaderNode** tab and assign the **diffuse shader** that you just imported.
7. Now go to the **Shader** tab:
    a. For Texture 0, it should already be pointing to your **diffuse** texture.

b. For Texture 1, choose your **specular** texture in the dropdown.
c. For Texture 2, choose your **environment** texture in the dropdown
d. For Texture 3, choose your **normal** texture in the dropdown.

FR Settings

You can adjust the settings in the effect node:

1. **Fresnel Falloff** controls how much of the surface reflects the environment. Set it to zero if you want chrome (yuck). Or around 1.5 for a snooker ball. Or around 3.0 for anything else.
2. **Fresnel Strength** controls the intensity of the reflection. How bright do you want the reflections? Keep it at 1.0 most of the time.
3. **Normal Strength** control the intensity of the normal map. You ought to make changes to the normal map outside of tOG. But if you're in a pickle, try adjusting it up to 2.0 or 3.0 to ramp up the bumpiness. Or scale it down to 0.0 for a totally smooth surface.
4. **Ambient Environment** adds a bit of diffuse colour to the object, based on the colours in the environment. You can leave it off in most situations. Set it to around 0.2 if you have a very dark object that needs brightening up in the blacks.
5. **Env Rotation** allows you to rotate the environment map to match the real environment around it. You ought to link this to a separate transform node, so you can affect multiple environments from one source.

Skybox

It can sometimes help to include a Skybox to see how your shader looks in relation to the environment.

- Import the blurred environment map.
- Create a sphere, scale it to 100,100,100. Assign the blurred environment map shader. Push it to the top of the Scene Graph, so it's behind everything else in the scene.
- In the shader State tab, turn off Lighting and Backface Culling.

Troubleshooting: The reflections are too strong

First, try blurring the environment map.

Second, try darkening the shadows and midtones of the environment map and just leaving the highlights.

Third, consider adjusting your spec map to dim the reflections in specific areas of the mesh.

Troubleshooting: The mesh looks black (or otherwise weird)

If you are using an imported mesh (from an FBX), it's possible that the mesh is missing some important information called tangents and bitangents (or binormals). These are normally defined during export but sometimes the exporting software doesn't export them properly. However, there is a way to compute the tangents and bitangents within tOG Edit.

1. Click on the geometry in the scene graph.
2. Click **Compute TBN Matrix**.
3. Click on the **Save icon** in the Geometry pane.
4. Enter the name of the mesh with the suffix "_TBN", e.g., if the mesh was called Cube, save it as "Cube_TBN".
5. **Save the project and close it.**
6. Reopen the project and the graphic.
7. If this hasn't fixed your problem, please email support@rtsw.co.uk with details of your problem (a screenshot, the project files and details of your version of tOG).

Applying PBR in tOG

A step by step guide to exporting and importing for physically based rendering (PBR).

Preparing the Images

Normally this process is handled by your PBR image creation software (Quixel Suite or Substance Painter). However, for artists without access to that software, here is how to prepare the images for tOG:

1. **Name your images nicely before import.** Be clear. Make sure they're unique—try using a simple naming format like "Graphic_Mesh_Map.ext", e.g., "Formation_Plinth_Normal.png."
2. **If you don't have a normal map**, create a PNG that's 128,128,255 and call it "Generic_Normal_Blank.png".
3. **Combine your ambient occlusion (ao), roughness and metallic maps together**:
    a. **Create a new image**. Call it "Graphic_Mesh_AOMetalRough", e.g., "Formation_Plinth_AOMetalRough".
    b. Go to the **Channels** window.
    c. Select the "**Red**" channel.
    d. Open your **AO image**, select all, copy it and **paste it into the Red channel** of your new image.
    e. Select the "**Green**" channel.
    f. Open your **Metallic** image, select all, copy it and **paste it into the Green channel** of your new image.
    g. Select the "**Blue**" channel.
    h. Open your **Roughness** image, select all, copy it and **paste it into the Blue channel** of your new image.
    i. Select the RGB channel.
    j. Save the image.

Applying PBR in tOG

1. **Import** the images:
    a. Albeedo (PNG)
    b. AO_Metal_Rough (PNG)
    c. Normal (PNG)
    d. Environment (HDR)
2. In the **Images tab**, tick …
    a. **Create Shader For Each Image Or TMV**
    b. **Shader Prefix** (enter an appropriate prefix … this helps organise your shaders later).
    c. Hit **Import**.
3. Add an **FX node** to the scene graph, above the model(s) that you wish to affect.
4. Choose the **PBR_Metal_Rough_Spherical** from the dropdown list in the FX node.
5. In the **shader** for the geometry, populate the texture slots as follows:
    a. tex0 - **Albedo** colour map.
    b. tex1 - HDR spherical **Environment** map.
    c. tex2 - **AO_Metal_Roughness** map.
    d. tex3 - **Normal** map.
6. **If your mesh looks black**, see Troubleshooting: The mesh looks black (or otherwise wierd) in the FR section above.

PBR Settings

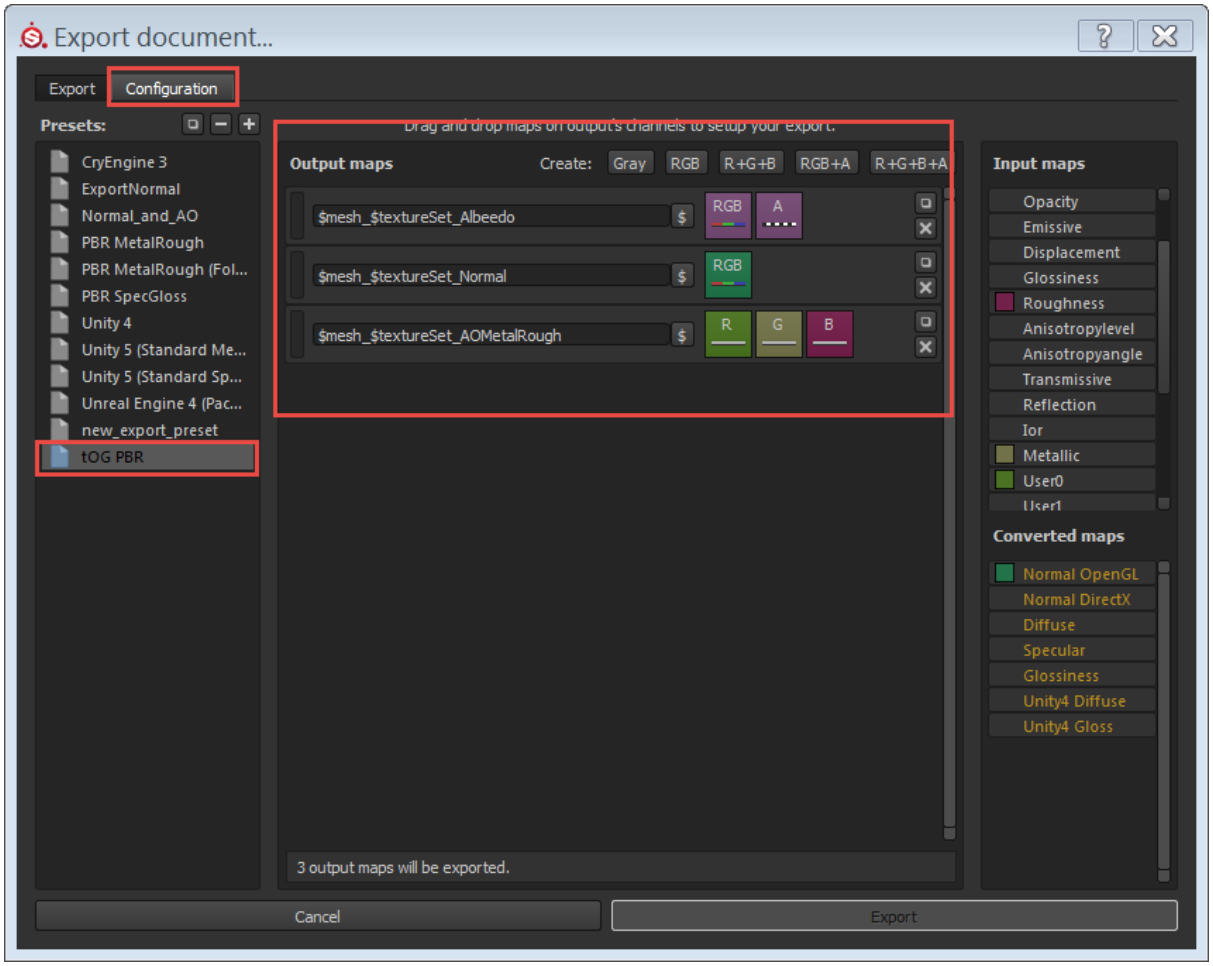
You can adjust the settings in the effect node:

1. **nbSamples** should always be set to 16.000.
2. **Env Expsoure** controls the exposure of the environment lighting. Set it to 0.5.
3. **Env Rotation** allows you to rotate the environment map to match the real environment around it. You ought to link this to a separate transform node, so you can affect multiple environments from one source.

Extra: Substance Painter Export

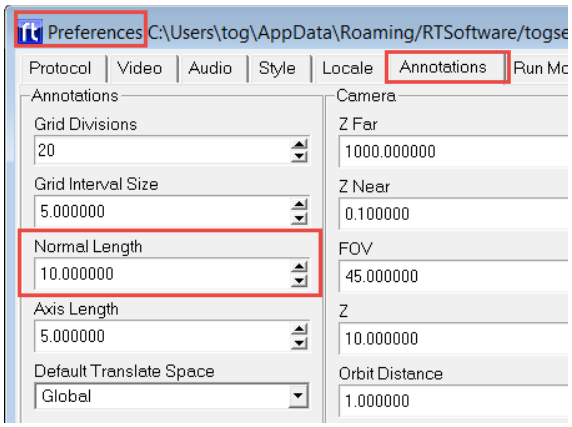
If you are using Substance Painter, you can export the images as follows:

1. In the TextureSet settings, click the + button beside Channels and add a new user0 Channel.
2. Change the format to L8.
3. Create a fill layer.
4. Turn off all channels in the fill except for the User0 channel.
5. In the Textures tab in the shelf, find the AO map. Drag this into the new fill layer's user0 channel.
6. Use Export all Channels from the File menu. Go to the configuration tab.
7. Create a new configuration by hitting the + button in the configuration tab (see the image below).
    a. It will be called "new_export_Preset" in the Preset list on the left. Rename this now to "tOG PBR".
    b. Create an RGBA output (click RGB+A) and drag the BaseColor (on the right) into it the RGB. Select RGB Channels.
    c. Drag the BaseColor into the A and select A channel.
    d. Rename it from "RGB Alpha" to $mesh_$textureSet_Albeedo.
    e. Now add an RGB output (click RGB) and drag the Normal OpenGL (on the right) into it the RGB. Select RGB Channels.
    f. Rename it from "RGB" to $mesh_$textureSet_Normal.
    g. Lastly add an RGB output (click R+G+B) and drag the User0 (on the right) into it the R. Select Gray Channel.
    h. Drag the Metallic (on the right) into it the G. Select Gray Channel.
    i. Drag the Roughness (on the right) into it the B. Select Gray Channel.
    j. Rename it from "R+G+B" to $mesh_$textureSet_AOMetalRough.
8. Go back into the Export tab and change the output path to a suitable location.
9. Also change the output format to png (16 bits max).
10. Change the Config to your new "tOG PBR".
11. Set the output size to 2048 x 2048 (or higher, if desired).
12. Hit Export.


Export document...
Export
Configuration
Presets:
CryEngine 3
ExportNormal
Normal_and_AO
PBR MetalRough
PBR MetalRough (Fol...
PBR SpecGloss
Unity 4
Unity 5 (Standard Me...
Unity 5 (Standard Sp...
Unreal Engine 4 (Pac...
new_export_preset
tOG PBR
Drag and drop maps on output's channels to setup your export.
Output maps
Create:
Gray
RGB
R+G+B
RGB+A
R+G+B+A
$mesh_$textureSet_Albeedo
RGB
A
$mesh_$textureSet_Normal
RGB
$mesh_$textureSet_AOMetalRough
R
G
B
3 output maps will be exported.
Input maps
Opacity
Emissive
Displacement
Glossiness
Roughness
Anisotropylevel
Anisotropyangle
Transmissive
Reflection
Ior
Metallic
User0
User1
Converted maps
Normal OpenGL
Normal DirectX
Diffuse
Specular
Glossiness
Unity4 Diffuse
Unity4 Gloss
Cancel
Export

Extra: Fixing Tangents and Bitangents

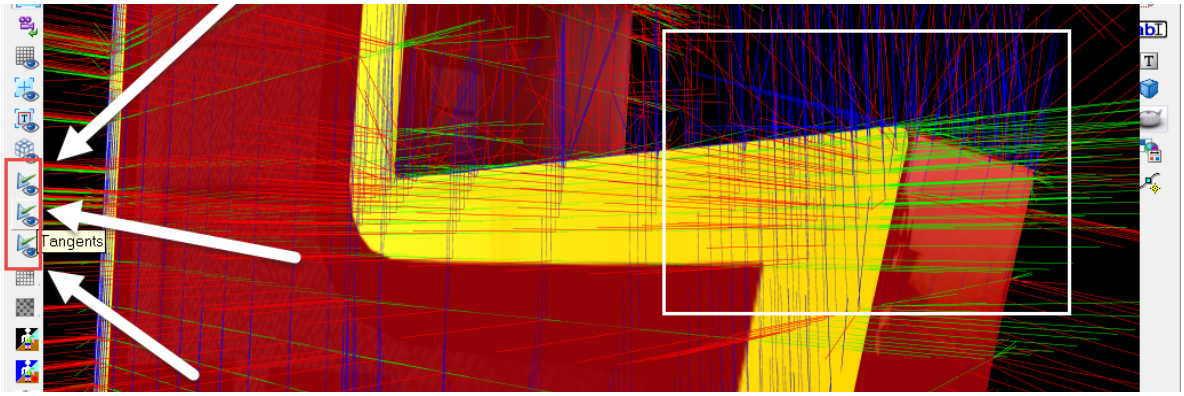
In tOG Edit 4.6.2, it is possible to check for binormals and tangents in the viewer.



- In the **Preferences** -> **Annotation** tab. Set the **Normal Length** to something like 10.
- Enable **normal/bitangent/tangent in the viewer** (on the left, see image below) and look for red, green and blue lines.

If you don't see green or blue lines, your model is missing important information (tangents and bitangents). This will limit the type of shader you can apply to your model.

N.B. At the time of writing, Cinema does not export bitangents and tangents.



If you don't have tangents and bitangents, we'll need to manually generate these within tOG:

1. Click on the geometry in the scene graph.
2. Click **Compute TBN Matrix**.
3. Click on the **Save icon** in the Geometry pane.
4. Enter the name of the mesh with the suffix "_TBN", e.g., if the mesh was called Cube, save it as "Cube_TBN".
5. **Save the project and close it.**
6. Reopen the project and the graphic.
7. If this hasn't fixed your problem, please email support@rtsw.co.uk with details of your problem (a screenshot, the project files and details of your version of tOG).