Innovations Report

Experimentation with Methods to Create Realistic Procedural Textures

Sally Wilson

Contents

Introduction

The focus of this report will be the discussion and experimentation of a current popular texturing method. Particular focus will be paid to the process of creating procedural textures using complex node networks in the Maya Hypershade. According to David S. Ebert:

> *"We consider procedural techniques to be code segments or algorithms that specify some characteristic of a computer generated model or effect."*
>
> - David S. Ebert, Texturing & Modelling A Procedural Approach, 1998[1]

Employing the commonly used hypershade feature in Maya as a method to create these 'code segments' is an easy way for modern day texture artists to colour their scene. Therefore, studying this particular process is a relevant way to explore a widespread procedural texture method. This report aims to examine how close to reality a surface can be pushed before the introduction of hand painted texture maps becomes necessary for realism.

Procedural texturing is an area which I currently know little about and would like to research further. I feel it is one of the main areas in which the art and maths side of this subject really collide. I intend to educate myself to know when it is best to use one method over the other and when a combination of the two is necessary. I predict that it is fundamental for any good texture artist to have an understanding of the maths involved with the procedural process.

Selecting a Subject Matter

Mathematics has always had a strong role to play in certain fields such as physics, but recently it has also been found to be applicable in biology. Genetic design that some originally thought to be due to part chance is actually the outcome of mathematical dynamics. The most noticeable example of this genetic design is that of the pattern on the coats or skins of animals.

Figure 1 Balancing frogs t-shirt design by www.teezz.co.uk

Animal skins range tremendously in design, variety, colour, shape and pattern. Some animals are spotty, some stripy, some plain and some a combination of them all. The specific pattern an animal may develop on its skin is ultimately down to a mathematical equation written by nature. This equation deals with the way in which two different chemical products react and are reproduced across the skin. Any particular patch on an animal's skin could be affected either by a chemical which

[1] Texturing & Modelling A Procedural Approach, Copyright Academic Press, 1998

incites the production of melanin and therefore colouring the skin, or one which inhibits this production.

Tropical frogs in particular demonstrate a very wide range of colours and designs within one species. Their bright skin designs are not only attractive but can be very detailed and specific to that subspecies of frog, allowing us to easily identify them. If frog skin patterns were formed by a mathematical equation in the first place, it follows that creating a texture for them should be possible procedurally. I have therefore decided to use tropical frogs for my subject matter as I feel they will allow me the freedom to work with a variety of colours, patterns and surfaces all within the same theme and on the same model.

Figure 2 Dendrobates azureus (Blue Poison Dart Frog), at home in the Southern Surinam.[2]

Researching Skin Shaders

To achieve the greatest understanding of the procedural texturing process it is important to consider the work and concepts of previous texture artists. I have also found that there is an extremely limited selection of resource material written for the Maya Hypershade making it necessary for me to gain the majority of my knowledge in this area from looking at the work of previous artists. An evaluation of the strengths of these existing shading networks can then be made and improved upon where possible. For this reason, this report will first look into the good and bad features of some free online reptile skin shaders created for Maya.

Wet1.0 Shader by Hwee Kok[3]

I began by looking at some free online skin shaders to get an idea of how procedural methods can be used to get the desired results. First, I had a look at the Wet1.0 shader by Hwee Kok.

- Wet 1.0
- This is a Wet look shader created by Hwee Kok...
- 
- UPDATED: 10/18/00
- FILE SIZE: 3 KB
  DOWNLOADED: 27,186
  HOMEPAGE: http://cade.scope.edu
  LICENSE TYPE: Freeware
  OWNER: admin Author Name: Hwee Kok
- DETAILS: This is a Wet look shader created by Hwee Kok the course manager of City University of Hong Kong School of Continuing and Professional Education.

[2] Image Source: www.oregonzoo.org
[3] Downloaded from www.highend3d.com

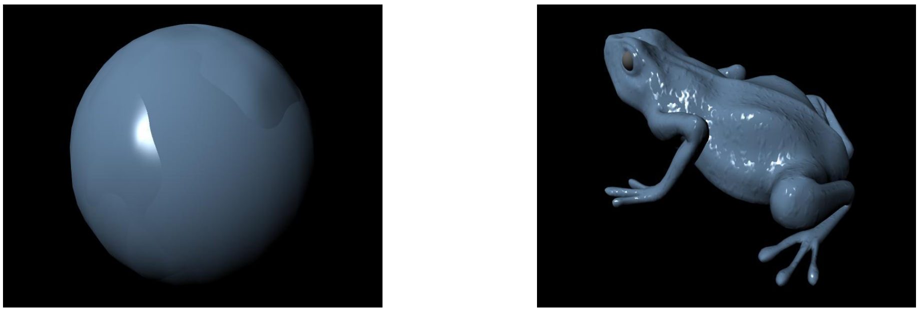


Figure 3 Sphere and a frog textured using the Wet1.0 shader by Hwee Kok

Kok has used a phong shader for the final output texture. The phong gives a better resolution of specular reflections than a standard Lambert which is ideal for this wet shader. Phong shading is a linear interpolation method by which for any polygon, we evaluate the vertex normal vectors, and thus we can derive a normal vector for each point or pixel on the polygon that is an approximation to the real normal on the curved surface approximated by the polygon. In this example, he has mapped a flat blue texture into the phong for the colour attribute which I feel is not very realistic. All surfaces have subtle variations in colour and hue, especially organic or wet ones. I suggest that this shader would be more effective had he used a surface with slight colour intensity variations.

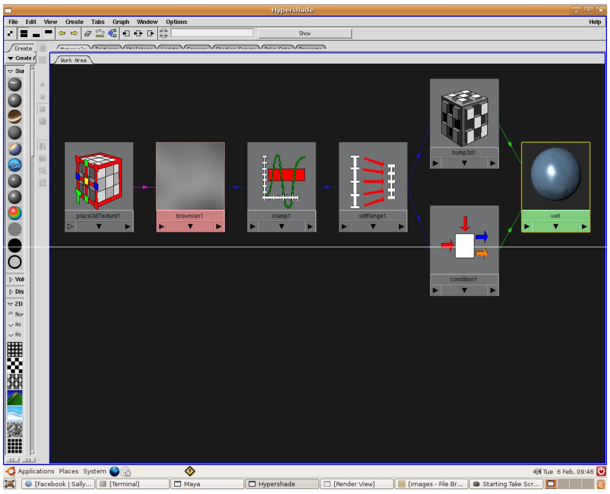


Figure 4 Shading Network of the Wet1.0 shader by Hwee Kok

A brownian texture has been used to get the subtle bump effect on the surface, simulating water droplets. This low contrast texture was named after the scientific term ‘brownian motion’[4]. It is the term used to describe the random motion of particles as they bump into each other. This brownian shader has been connected through a clamp function which is used to confine an increasing, decreasing, or randomly changing number to a range of values. Using this node will ensure that the intensity of the water reflections are kept realistic. Although this works quite well, it only really looks good at a distance. Close up, this shader looks as if a second piece of geometry is protruding the surface.

[4] *Brownian motion* (named in honour of the botanist Robert Brown)

Scaly1.0 Shader by Hwee Kok[5]

- Scaly 1.0
- This is a scaly shader created by Hwee Kok the...
- 
- UPDATED: 10/19/00
  FILE SIZE: 3 KB
  DOWNLOADED: 26,408
  HOMEPAGE: http://cade.scope.edu
  LICENSE TYPE: Freeware
  OWNER: admin Author Name: Hwee Kok
- DETAILS: This is a scaly shader created by Hwee Kok the course manager of City University of Hong Kong School of Continuing and Professional Education.

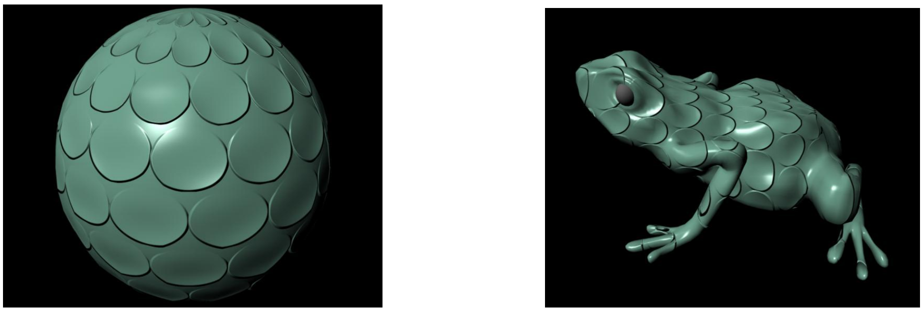


Figure 5 Sphere and a frog textured using the Scaly1.0 Shader by Hwee Kok

Again, a phong has been used for the output shader in this case, and similarly to before, a flat colour has been used. From looking at these examples, I know that the use of a phong shader could be ideal for the output shader on my frog's skin texture but I will need to create a colour input with more surface variation. The pattern in this shader has been created through the clever use of two ramps and UV coordinates assigned as bumpValue. It is a process whereby the circular pattern in the ramp is used to determine the normalCamera attribute of the final phong. A ramp is a colour gradient in which the colour value changes from one value to another across the coverage of the image. Plugging the outUV value of a place2dTexture node into the uvCoord value of a ramp, will mean that the pattern is affected by its position in relation to the camera. A circular ramp has been used to simulate the scales although I don't think these are very realistic. Real life scales over-lap, and these

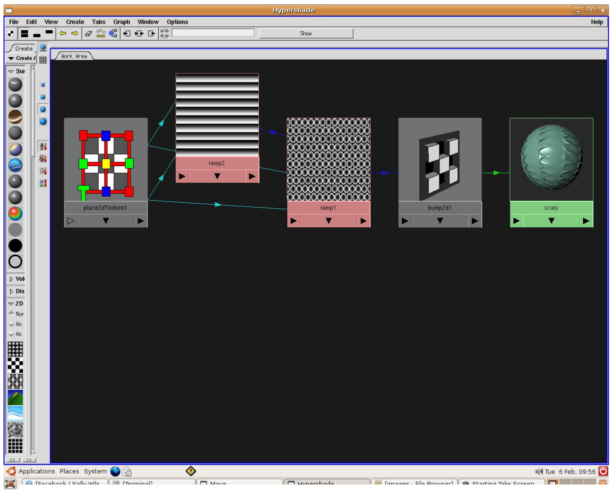


Figure 6 Shading Network of the Scaly1.0 Shader by Hwee Kok

[5] Downloaded from www.highend3d.com

are simply circles positioned neatly side by side. It would have been more appropriate to introduce an element of randomness, perhaps using the noise value to deregulate the circles.

Reptile1.0 Shader by Nicool[6]

- Reptile 1.0
- Bumped shader for reptile. Wonderful for Snake...
- FILE SIZE: 2 KB
  DOWNLOADED: 24,751
  HOMEPAGE: http://www.nicool.fr.st
  LICENSE TYPE: Freeware
  OWNER: nicool Author Name: Nicool
- DETAILS: Bumped shader for reptile. Wonderful for Snake Model!

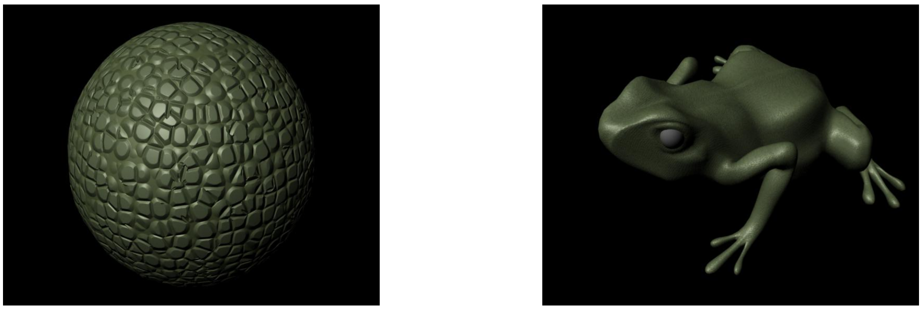


Figure 7 Sphere and a frog textured using the Reptile1.0 Shader by Nicool

Nicool has chosen to use a blinn shader for this reptile texture. He has created a very simple network for this skin, using only a leather texture for both the bump and colour inputs. The final texture resembles more a leather material than any realistic creature's skin. This texture seems very plastic, mainly because it has no value for diffuse or specular, let alone incandescence. These three values are very important for creating realistic textures, particularly for living creatures.

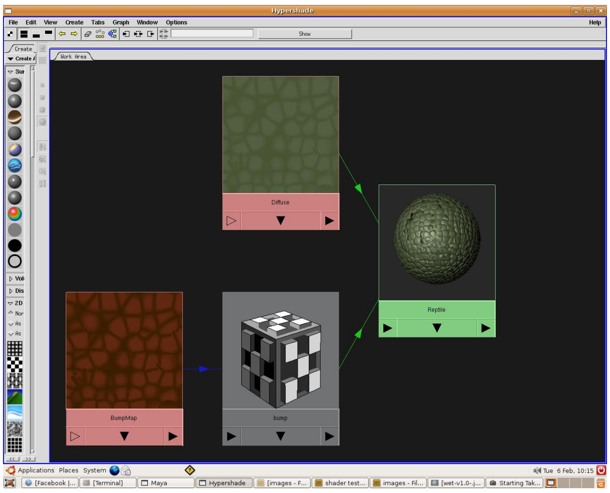


Figure 8 Shading network of the Reptile1.0 Shader by Nicool

[6] Downloaded from www.highend3d.com

Most objects do not emit any light of their own. Instead, they reflect part of the light that they receive and absorb the rest. The intensity of light that is reflected is dependant on the interaction between the incident light and the surface material. A surface will reflect coloured light when illuminated by white light and the coloured reflected light that we see is due to diffuse reflection. A surface will rarely scatter light equally in all directions giving us many variations of intensity and colour for diffuse depending on the object. This means that a good diffuse map is very important when trying to make anything look photorealistic because it describes the way objects interact with the light around them. Unlike diffuse reflection, where incoming light is reflected across a wide range of angles, specular reflection sends light in one clear path. According to Bill Fleming:

> *"…specularity is key to visualising surface texture…"*
>
> - Bill Fleming, Advanced 3D Photorealism techniques, 1999[7]

All objects demonstrate a certain amount of diffuse and specular reflection; it depends on the smoothness of the surface as to how much. It is the specular reflection that causes the small highlights we see in surfaces as the light reaches a certain angle.

<u>CrustySkin1.0 by Eric Luther[8]</u>

- Crusty Skin Shader 1.0
- Could be used for weird skin texture, changing...
- 
- UPDATED: 06/01/01
  FILE SIZE: 4 KB
  DOWNLOADED: 27,002
  LICENSE TYPE: Freeware
  OWNER: digismash Author Name: Eric Luther
- DETAILS: Could be used for weird skin texture, changing colours opens a lot of possibilities.

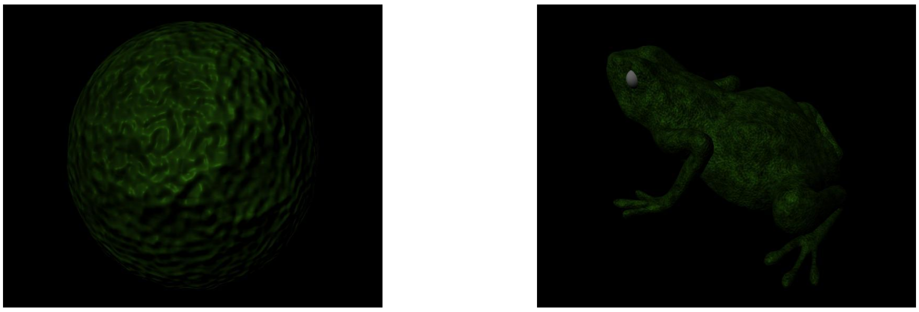


Figure 9 Sphere and frog textured using CrustySkin1.0 by Eric Luther

[7] Advanced 3D Photorealism Techniques, Copyright John Wiley & Sons, Inc, 1999
[8] Downloaded from www.highend3d.com

Luther, like Nicool, has also opted to use the blinn as the output shading surface. The blinn surface is an approximation of the phong. Its reflections are calculated in a similar way, but the blinn can produce more accurate models of reflected radiance than the phong. This blinn has only three attributes being procedurally calculated, a bump map, a colour map, and a specular map. All three of these attributes are mapped from the same node network ending in a leather node. The leather texture uses a 3D array of spheres to simulate 2D leather. Firstly, Luther has used a crater node for the density of the leather, a marble node for randomness of the leather pattern, another crater node to define the spottiness of the leather texture as well as the colour and finally, a brownian node for the crease colour. The result is a very organic and random looking, highly varied procedural texture. The effect is certainly reminiscent of reptile skin but no specific pattern is clear which would be necessary for the poison arrow frog skin.

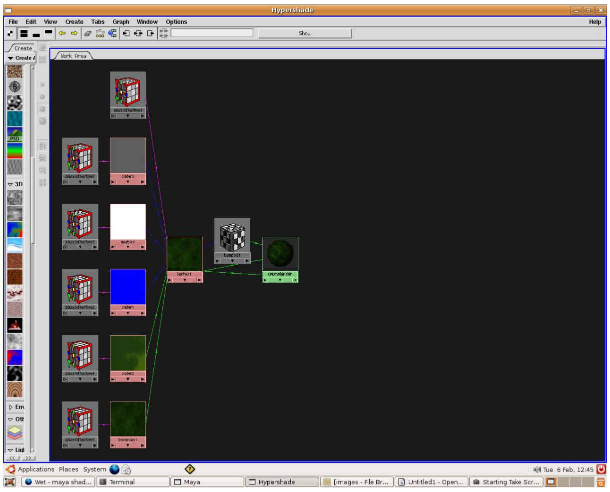


Figure 10 Shading network of CrustySkin1.0 shader by Eric Luther

<u>WetReptileSkin1.0.0 by Nenad Jalsovec[9]</u>

- Wet Reptile Skin 1.0.0
- Moist skin of frogs, toads, geckos ...
- 
- UPDATED: 11/01/05
  FILE SIZE: 8 KB
  DOWNLOADED: 10,853
  LICENSE TYPE: Freeware
  Maya Versions: 6.x, 7.x
  OWNER: failure Author Name: Nenad Jalsovec
- DETAILS: Moist skin of frogs, toads, geckos etc. Layered shader that uses 2 layers. Subsurface scattering is faked using mapped incandescence...

[9] Downloaded from www.highend3d.com

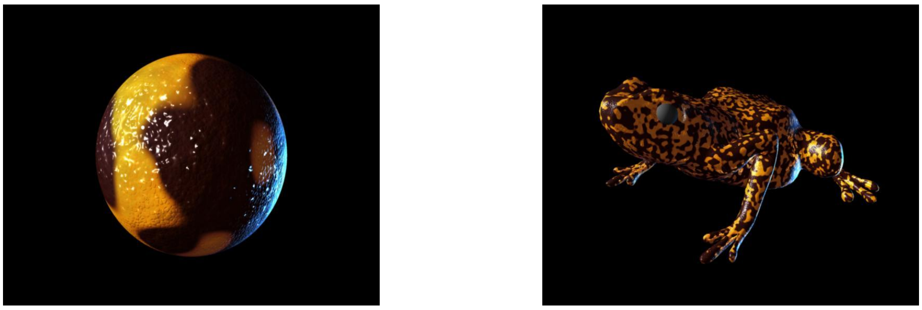


Figure 11 Sphere and frog textured using the WetReptileSkin1.0.0 shader by Nenad Jalsovec

This is the most accurate reptile skin shader example so far, and it is also the most complex. It uses two layers to achieve the wet look on top of the skin texture. The basic colour has been created using a volumeNoise node plugged into a remapRamp node in brown and yellow. Incandescence has been cleverly used to fake subsurface scattering by multiplying the colour channel's remapRamp node with a ramp shader to vary the colour distribution slightly.

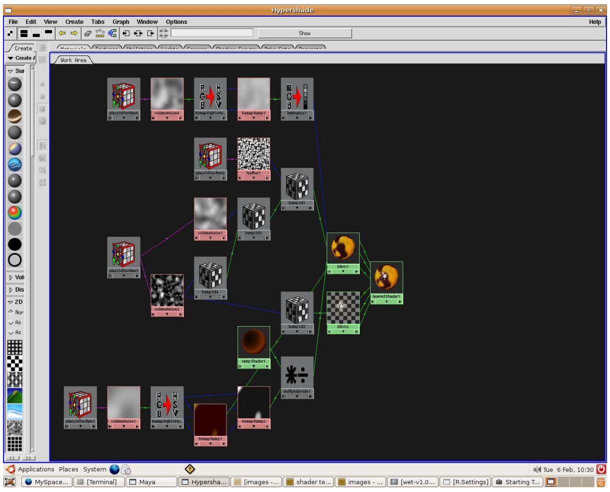


Figure 12 Shading network for WetReptileSkin1.0.0 shader by Nenad Jalsovec

Incandescence occurs when a heated solid gives off light of its own. Subsurface scattering is the process by which light can penetrate the surface of a translucent object which we see most noticeably in the human skin. Using incandescence instead of the standard SSS shader which is built into most 3d packages can be much more cost effective. It creates a layered look to the surface, simulating the translucence we see of skin cells in real life. This shader is also the first we've looked at to use diffuse. This channel simulates the reflection of light that can be seen from an irregular or rough surface. A volumeNoise and remapRamp node has been used with a luminance node to get this subtle light reflection effect, essential when simulating real life. The luminance utility is used to convert colour into greyscale, meaning each pixel can now be read as a value between 1 and 0 and so can be used in many more mathematical functions. Finally, the bump map has been formed from a Billow volumeNoise node, perlin VolumeNoise node and a leather node. Using the three of these different textures at different intensities gives a subtly varied surface, more true to life than any of the previous examples.

Developing my Skills Learnt from Research

Before beginning texturing exercises I took a trip to the nearest zoo to spend the day observing the frogs in the Tropical House. This was essential to ensure I had plenty of reference photographs, footage and to study the environment. More of the photographs I took can be found in the appendix of this report.

Figure 13 Poison arrow frogs in their captive habitat, photographed at Marwell Zoo, Dorset.

After spending the day filming and photographing for research, I decided to attempt a procedural texture for a couple of test images. My plan was to recreate the skin textures for each of these sub-species of frog using a purely procedural approach. I was not going to allow myself to use any hand painted textures to aid the process. I chose the following two images as an exercise to test the knowledge I had gained from my research of the Maya Hypershade before attempting my final textures.

Figure 14 Yellow and Black Poison-arrow at Marwell Zoo, Dorset.

Common Name: Yellow and Black Poison-arrow Frog, Bumblebee Poison-arrow Frog.
Scientific Name: Dendrobates leucomelas.

Distribution: Venezuela and eastern Guyana.

Common Name: Blue Poison-arrow Frog, Blue Azureus.
Scientific Name: Dendrobates azureus.

Distribution: Sipaliwini Savannah of Southern Surinam.

Figure 15 Blue Poison-arrow Frog. Source: www.geocities.com/lizhatfield/az1c.jpg

- <u>Yellow and Black Poison-arrow Frog Skin</u>

Stage 1 – Creating the colour

Starting with a vRamp node, split between a dark brown and mustard colour at 0.4, a volumeNoise node is used to determine the U and V coordinate of the resulting texture. The effect is an apparently random looking distribution of mustard coloured spots on a dark brown surface.

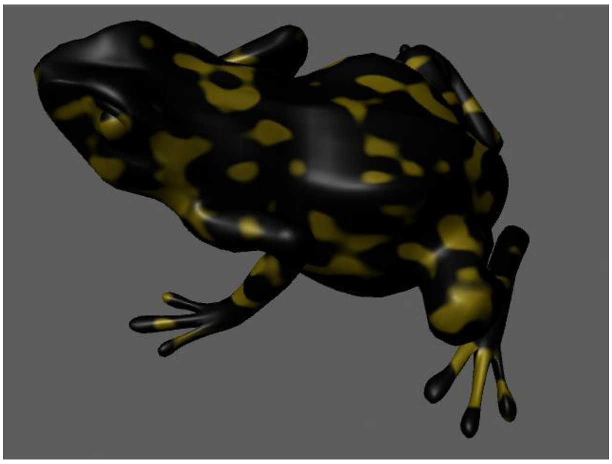


Figure 16 Stage 1 - Texturing the Yellow and Black poison arrow frog skin

Stage 2 – Creating the incandescence

Simply using the same vRamp set-up again for the incandescence gives a far too overly saturated black and white contrast.

Figure 17 Stage 2 - Texturing the Yellow and Black poison arrow frog skin

Stage 3

Using a new linearly interpolating ramp shader with three levels of dark/light/dark for incandescence instead of the two in the vRamp, gives a far more desired finished, although still not quite realistic enough. Here I have tried using a dark brown set at 0.0, mustard at 0.45 and dark brown again at 0.87.

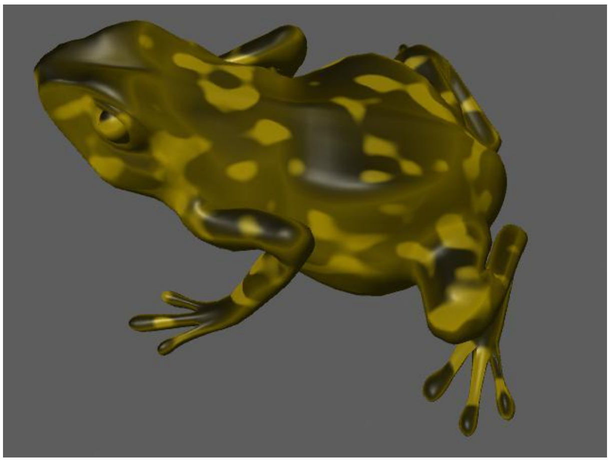
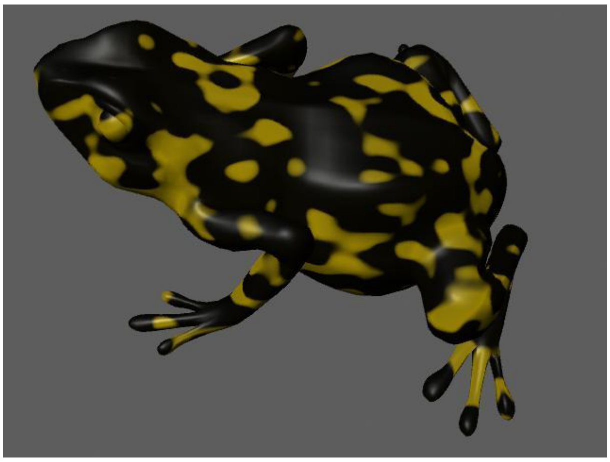


Figure 18 Stage 3 - Texturing the Yellow and Black poison arrow frog skin



Stage 4

The two previous attempts for an appropriate incandescence are still not quite right. However, multiplying these two levels of incandescence gives a more realistic highlight. See figure 19 left.

Figure 19 Stage 4 - Texturing the Yellow and Black poison arrow frog skin

Stage 5 – Creating the bump map

Now that a basic colour has been set, it is a good time to find a suitable bump texture. Starting with the volumeNoise node again, using a threshold of 0, amplitude of 1, ratio of 0.5, frequency ratio of 2.0, depth max of 2 and frequency of 8.55. The bump texture is far too severe and stippled.

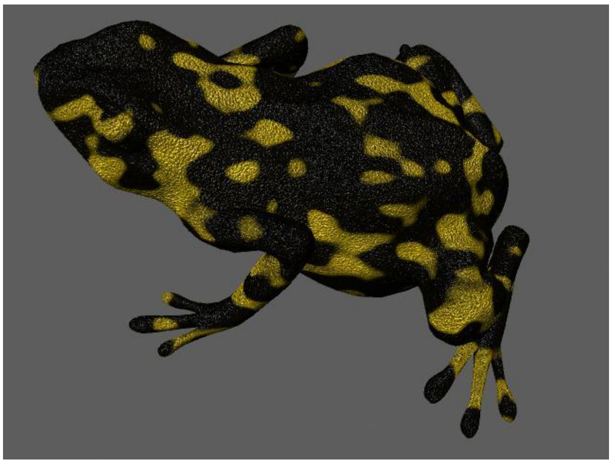


Figure 20 Stage 5 - Texturing the Yellow and Black poison arrow frog skin

Stage 6

By reducing the intensity of the bump down to 0.002 and plugging it into another volumeNoise node with more subtle levels produces a kind of multiplying effect. This new volumeNoise node has been set with values of 1 for depth max and a much lower frequency of only 5.0.

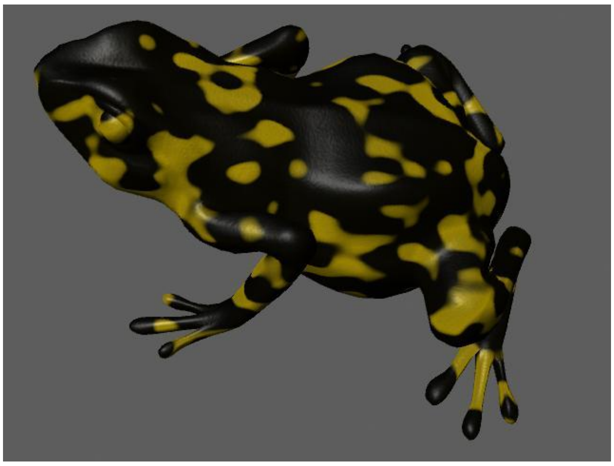


Figure 21 Stage 6 - Texturing the Yellow and Black poison arrow frog skin

Stage 7

Reducing the intensity of this bump map again to only 0.07 and then plugging it into yet another node will increase the subtlety and variation further. Using a leather texture node with cell size 0.2, density1, spottiness 0.2, randomness 0.7 and threshold 0.8 helps provide a more life like scaly reptile skin. Figure 22 below shows this bump map set to the default level of 1.0. Figure 23 also below, shows the final result after reducing this bump value down to 0.006 to provide a more convincing surface.

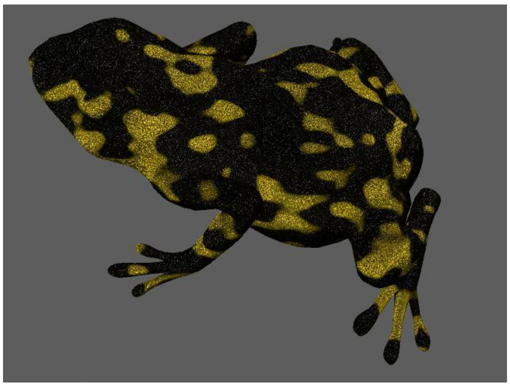


Figure 22 Stage 7a - Texturing the Yellow and Black poison arrow frog skin

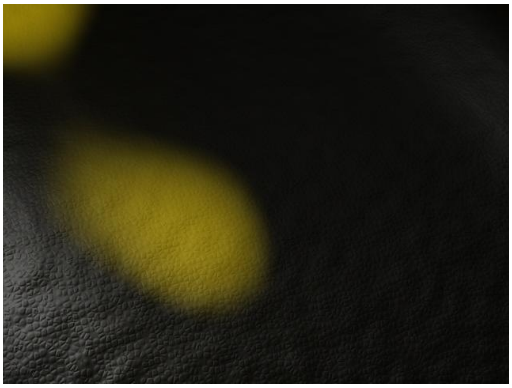


Figure 23 Stage 7b - Texturing the Yellow and Black poison arrow frog skin

Stage 8 – Creating the diffuse

Now that the skin has a reptile scale and a roughly accurate colour, adding some diffuse will help bring it closer to reality. Using the volumeNoise node, set threshold to 0.0, amplitude to 1.0, ratio to 0.6, frequency ratio to 2.0, depth max to 1.0 and frequency to 4.0. Using a luminance utility node to convert the outColour of the volumeNoise into a value useable for diffuse we can get this effect, which although more lifelike, is a little too stippled again.

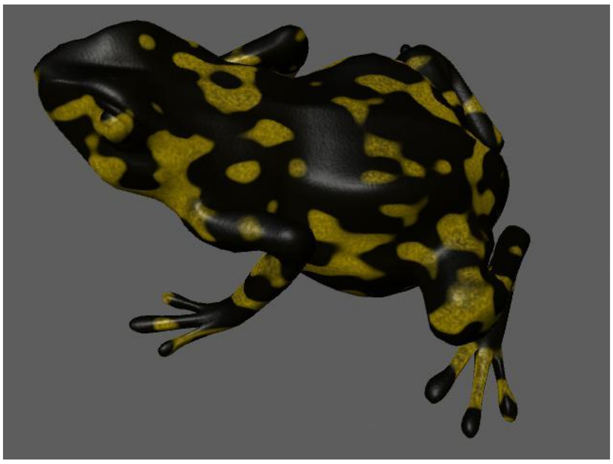


Figure 24 Stage 8 - Texturing the Yellow and Black poison arrow frog skin

Stage 9

Instead, using the Rgb-to-Hsv Utility node, we can map the Hsv values of the volumeNoise node into the U and V coordinates of a remapRamp node, whose values are a light grey and dark grey split evenly. Then using the luminance utility again to convert the outColour into an outValue for diffuse will create this less harsh mottled effect.

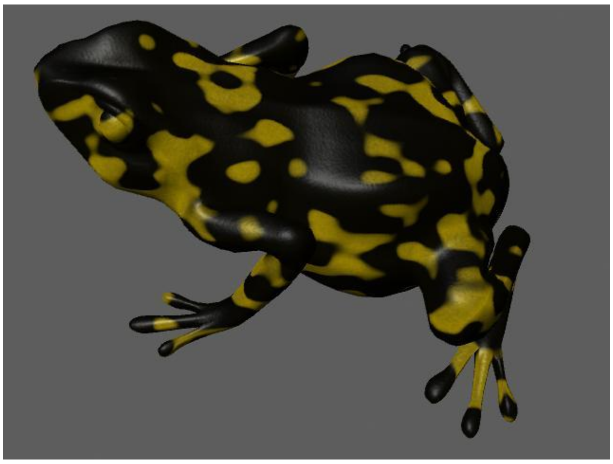


Figure 25 Stage 9 - Texturing the Yellow and Black poison arrow frog skin

Stage 10 – Creating the 'wet look'

To truly make this creature seem accurate, it needs to have a suitable wet and slimy skin. Using the layered effect I discovered from the 'Wet_Reptile' shader by Nenad Jalsovec, means I can create a wet shader separately and combine it with my previous one for the final effect. Using the volumeNoise node one more time as a starting point, setting values of threshold to 0.0, amplitude to 1.0, ratio to 0.5, frequency ratio to 2.0, depth max to 2 and frequency to 8.6. Plugging this node into the bump value of an orange blinn with transparency set very high can get a jelly like look. I have used a bump depth of 0.150 with diffuse set to 1.0, translucence set to 0.8 and translucence depth set to 0.5.

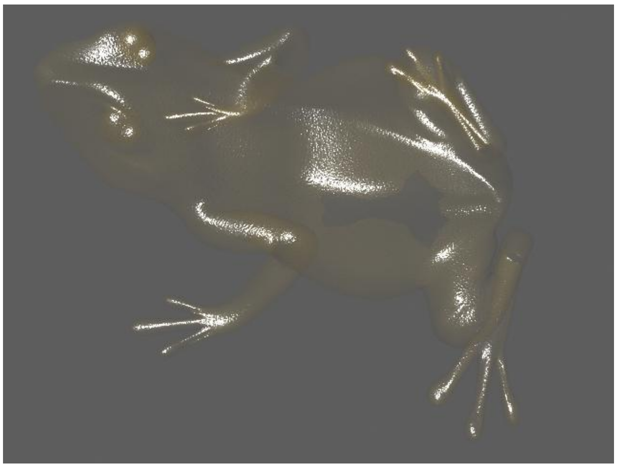


Figure 26 Stage 10 - Texturing the Yellow and Black poison arrow frog skin

Stage 11

Finally combining these two shaders using a layer node will give this effect shown below in Figure 27. I have connected the outColour, outGlowColour and outTransparency attributes of each of the previous shaders into this new layered one.

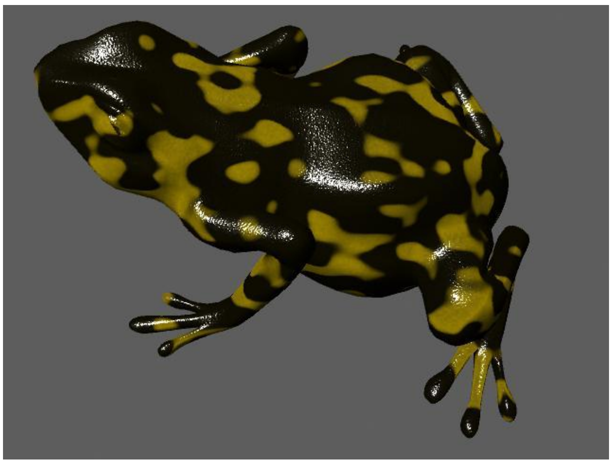


Figure 27 Stage 11 - Texturing the Yellow and Black poison arrow frog skin

- <u>Blue Poison-arrow Frog Skin</u>

Stage 1 – Creating the colour.

Start with a smooth circular ramp from black to white with a noise level of 0.3 and a noise frequency of 0.5. Setting the UV Coordinates to be determined by the place2dTexture OutUV and the filterSize to be determined by the place2dTexture outUvFilterSize gives a stippled circular pattern.

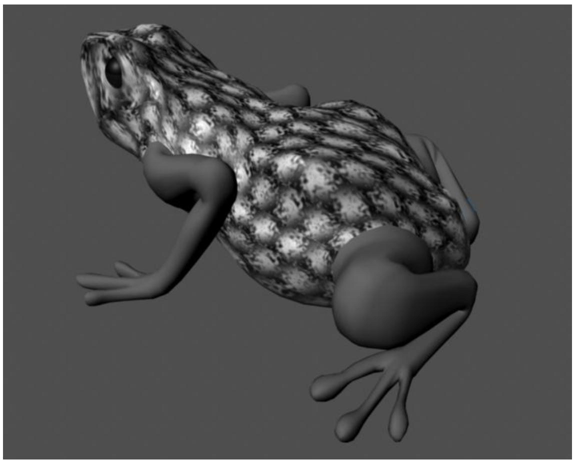


Figure 28 Stage 1 - Texturing the Blue poison arrow frog skin

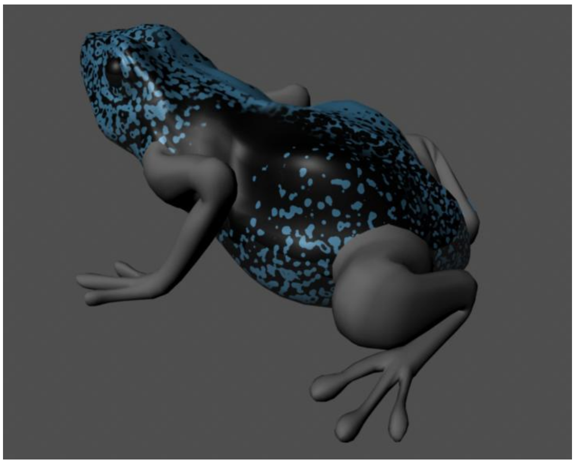


Figure 14 Stage 2 - Texturing the Blue poison arrow frog skin

Stage 2.

Using the outAlpha channel of a billow volumeNoise node as the u and vWave input on a blue and black linear vRamp gives a mottled black and blue texture, favouring the distribution of the blue to the outside of the object. Neither this, nor the previous colour is correct for the Blue Poison Arrow frog but the layout of colour on each is close. More work is needed, and a collaboration of the previous two colour examples, to achieve a colour map to more closely resemble a believable Blue Poison Arrow Frog skin shader.

Stage 3.

The distribution of the first stage mixed with the colour of the second is more appropriate for the Frog skin in question. Using another ramp node, with the outAlpha of the previous ramp mapped into the alphaGain channel, and the outColour of the previous volumeNoise node mapped into the colourValue of the secondEntryList will account for each of the previous stages to be included. Adding two other entryList values to the Ramp node, a blue above and a black below gives a good start to the colour map.

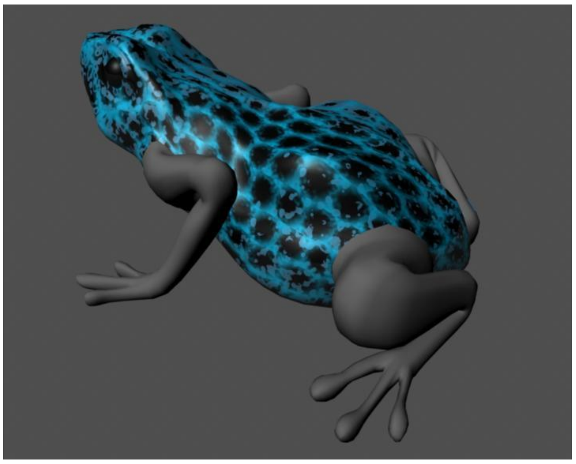


Figure 30 Stage 3 - Texturing the Blue poison arrow frog skin

Stage 4 – Creating the bump map.

Before much more work on the colour should be done, it is a good idea to introduce some bump texture. Similarly to the bump map on the Yellow and Black Poison Arrow Frog, it is a good idea to start out with a simple Billow volumeNoise node, threshold 0.0, amplitude 1.0, ratio 0.5, frequency ratio 2.0, depth max 2.0 and frequency 8.6. This creates a general all-over bump which is currently too strong and generic and will need to be worked into.

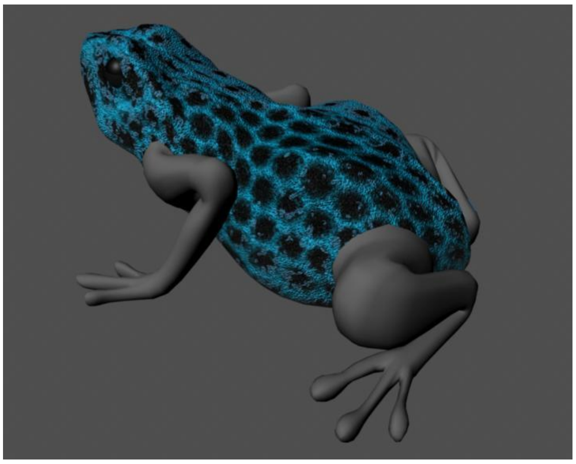


Figure 31 Stage 4 - Texturing the Blue poison arrow frog skin

Stage 5.

Reducing this billow volumeNoise node bump value from the default 1.0, down to only 0.002 will make it extremely faint. This can then be plugged into the normalCamera channel of another bumpMap, namely a perlin noise volumeNoise node, set to threshold 1.0, amplitude 1.0, ratio 0.5, frequency ration 2.0, depth max 1.0 and frequency 3.0. This is a much softer looking bump with a greatly reduced frequency.

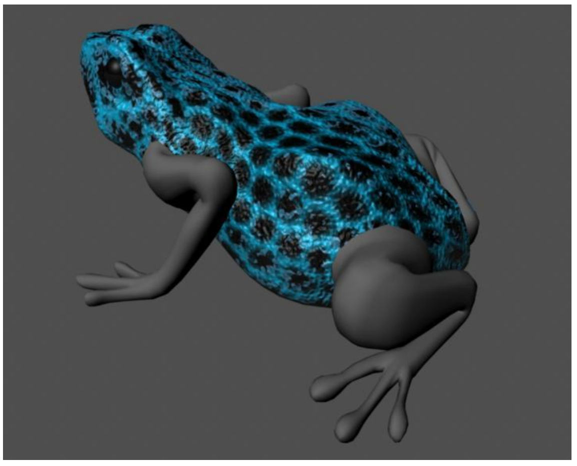


Figure 15 Stage 5 - Texturing the Blue poison arrow frog skin

Stage 6.

To gain even more refinement in this bump map, it helps to reduce this previous bump value to only 0.050, slightly higher than the first, and plug it yet again into another bump map. This time using a leather node set to cell size 0.7, density 0.5, spottiness 0.3, randomness1.0 and threshold 0.8. The result seen in the Figure 33 below shows the variations produced in the surface, although more appropriate, they are too severe. To finish off the bump, reduce the final map value to only 0.005, which can be seen in figure 34, also below.

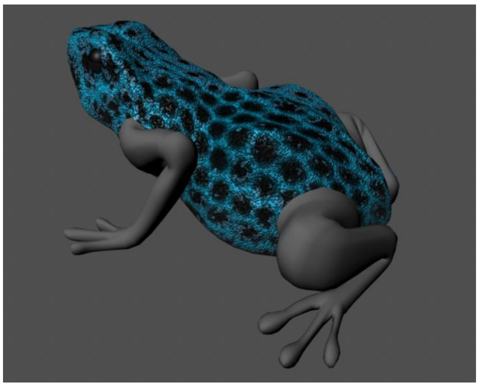


Figure 33 Stage 6a - Texturing the Blue poison arrow frog skin

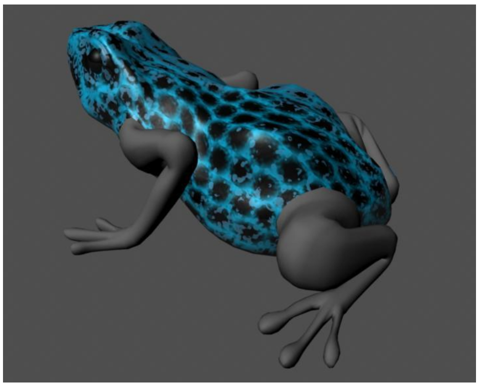


Figure 34 Stage 6b - Texturing the Blue poison arrow frog skin

Stage 7 – Creating incandescence.

To improve the colour of the shader and add depth to the skin, it can be effective to add incandescence. Using a 3 levelled linearly interpolated ramp shader in black/blue/black with the colour input set to 'brightness' will give the very exaggerated look shown right in Figure 35.

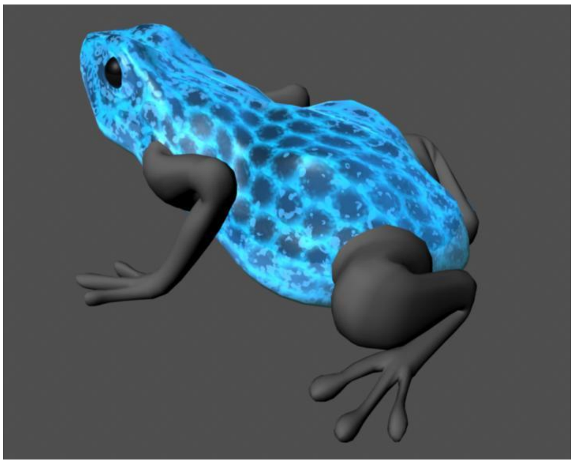


Figure 35 Stage 7 - Texturing the Blue poison arrow frog skin

Stage 8.

If you multiply the outColour of this ramp with the outColour of the ramp we used previously for the colour input of the shader, it will tone down this extreme glowing look to something more realistic and in keeping with the pattern of the skin. Figure 36 below shows the colour output of the above multiplication, and Figure 37 shows the result if you apply the output of this multiplication as the incandescence value of the blinn shader.

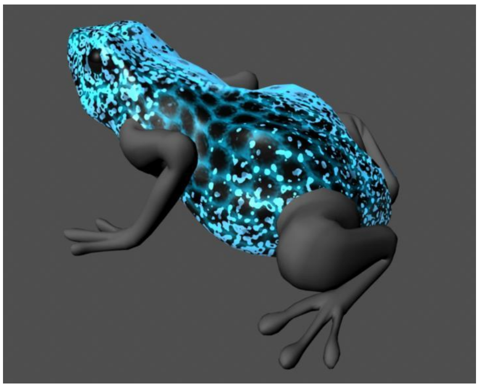


Figure 36 Stage 8a - Texturing the Blue poison arrow frog skin

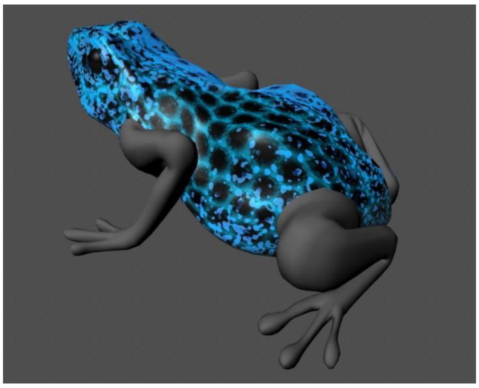


Figure 37 Stage 8b - Texturing the Blue poison arrow frog skin

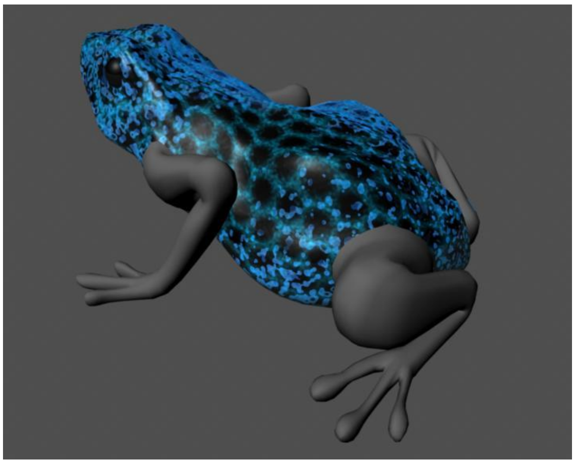


Figure 16 Stage 9 - Texturing the Blue poison arrow frog skin

Stage 9 – Creating diffuse.

A final element to the base texture which is essential for realistic surface shading is the diffuse channel. Diffuse gives the material the ability to reflect light in all directions. Here I have started out with another perlin noise volumeNoise node with a frequency of 4. By plugging this into an Hsv-to-Luminance utility node, I can get an output that is useable by the diffuse channel function.

Stage 10.

Again, when working with organic materials, particularly living creature's skin, subtlety and variation is key. By plugging the output of this perlin noise volumeNoise node into an Hsv-to-Rgb utility node instead, I can convert the output to be plugged into the u and v coordinates of a remapRamp node. This allows me to faintly affect the colour balance of the diffuse map as the remapRamp node allows me to limit the lightest and darkest areas. The outColour of this new node network must also be passed through an Hsv-to-Luminance utility node before it can be assigned to the diffuse channel of the final blinn shader.

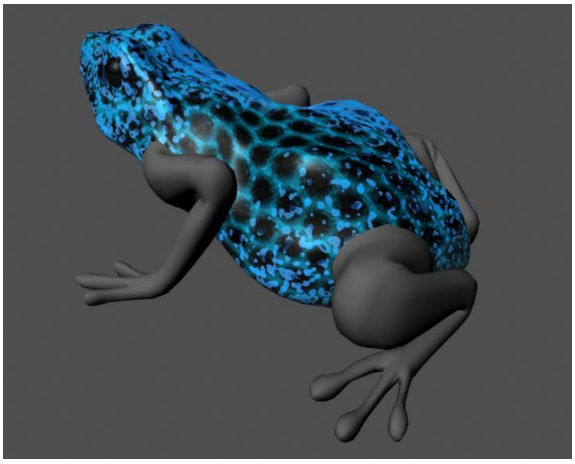


Figure 39 Stage 10 - Texturing the Blue poison arrow frog skin

Stage 11 – Creating the 'wet look'.

This was successful before when creating the yellow and black poison arrow frog texture so I decided to stick with the same network again. As before, a layeredShader has been used for the final material to combine the two. Figure 40 below shows the wet shader by itself, and Figure 41 shows the result when combined with the previous shader.

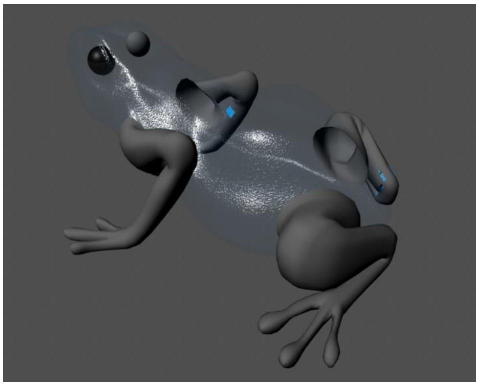


Figure 40 Stage 11a - Texturing the Blue Poison arrow frog skin

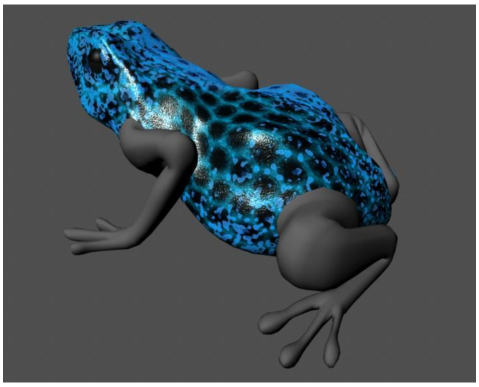


Figure 41 Stage 11b - Texturing the Blue Poison arrow frog skin

Stage 12 – Creating the texture for the legs.

Unlike the yellow and black poison arrow frog skin texture, this variation of the species seems to have more of a distinct dissimilarity in the surface of the legs to that of the main body. For this reason I felt it would be more appropriate to create a separate shader for this. For the base colour of this shader I have made use of the samplerInfo utility node. By connecting the facingRatio to the vCoord of a standard smooth linearly interpolated vRamp, we get a smooth colour which shifts slightly as the normals begin to point away from the camera. This means that the edges of the shape can be coloured differently to the centre. Since the blue poison arrow frog seems to have a lighter shade of blue around the edges of its legs and feet, this is a very useful connection to make. For continuity with the main body, I have used exactly the same node network for the diffuse and bump maps. Figure 42 below shows the result of using the facingRatio to determine the vCoord of the ramp and Figure 43 shows this colour with the original bump network included.

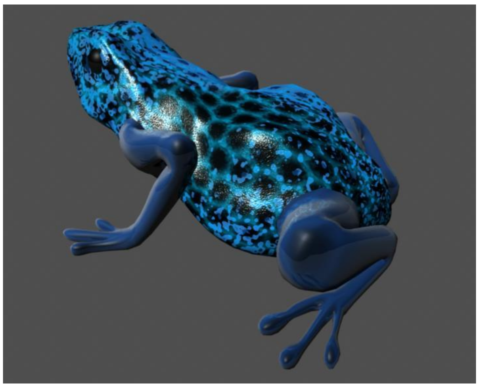


Figure 42 Stage 12a - Texturing the Blue Poison arrow frog skin

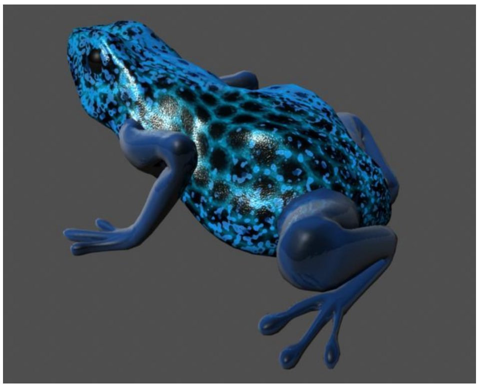


Figure 43 Stage 12b – Texturing the Blue Poison arrow frog skin

Stage 13 – Creating specular colour for the legs.

Unlike the main body, I have not used Incandescence to add depth to the skin, but Specularity instead. Incandescence determines the colour and brightness of light that a material appears to be emitting, whereas the specular colour of a surface determines the colour of the shimmering highlights on the surface. I used the same node setup as I did with the incandescence on the body before, but passed it through a blinn node first. By reducing the transparency of this separating blinn, I was able to reduce the intensity of the overall specularity.

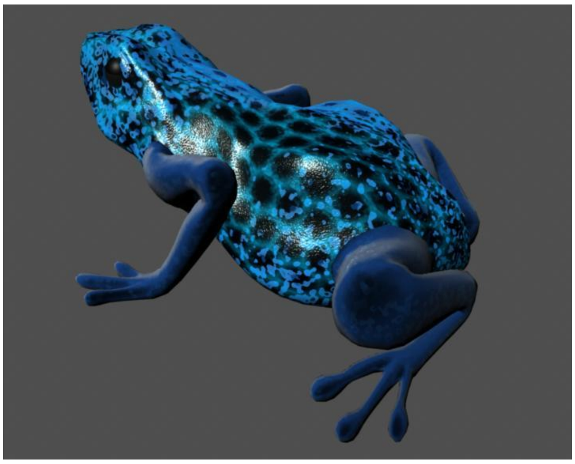


Figure 44 Stage 13 - Texturing the Blue poison arrow frog skin

Stage 14 – Creating diffuse for the legs.

Finally, adding diffuse exactly the same as before and adding the wet layer over the top gives the final result for my attempt at a shader for the blue poison arrow frog skin. Figure 45 and 46 below show the final result and an image of a real blue azureus poison arrow frog for comparison.

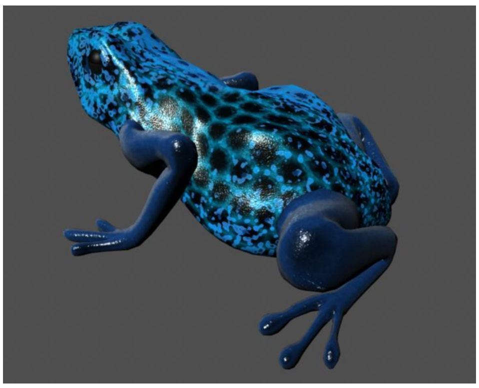


Figure 45 Stage 14 - Texturing the Blue poison arrow frog skin

Figure 46 Blue azureus frog.
Source: www.northwestfrogfest.com

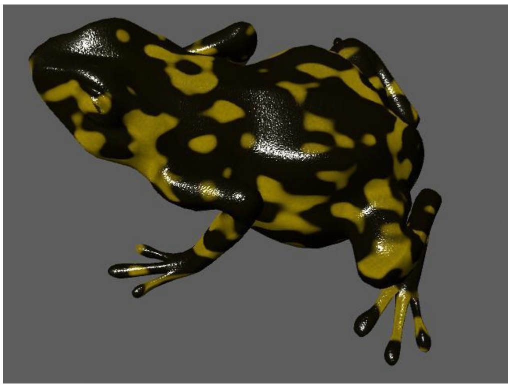


Figure 47 Final texture created for the yellow and arrow frog

Figure 48 Yellow and black poison poison arrow frog.
Source: upload.wikimedia.org

Looking back at the previous texture created for the yellow and black poison arrow frog, shown above in Figures 47 and 48, I can see now that it is not too accurate. The final texture will need to be brightened up and adjusted to be more specific before the final scene can be rendered out.

Environment Texturing Exercise

Before starting work on the final images for this report I wanted to have a go at creating a generic environment texture. I decided to create a rock texture similar to this one shown below in figure 49.

Figure 49 Blue Azureus Poison Arrow Frog. Source: www.redrobe.com

Stage 1 – Creating the colour.

Starting out with a rock shader, grain size 0.035, diffusion 0.716 and a mix ratio of 0.480 gives a sandier look than a rocky surface. Although, as shown right in figure 50, this is a good all over base texture to build on to.

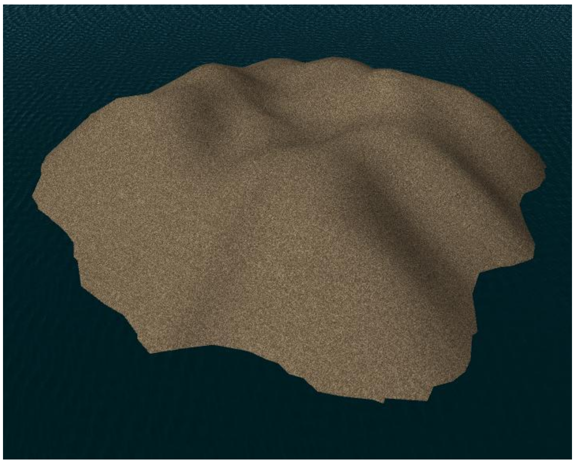


Figure 50 Stage 1 - Texturing the rock surface

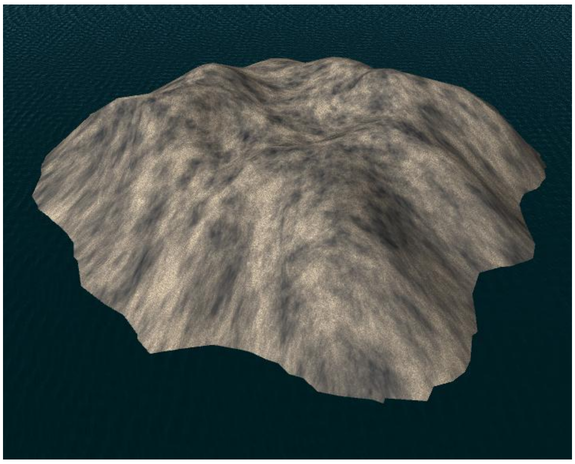


Figure 51 Stage 2 - Texturing the rock surface

Stage 2.

Creating a fractal node, and using the previous rock node to determine the colourGain we can achieve a more varied colour distribution. Fleming tells us in his book '3D Modelling & Surfacing' that:

> *"(Fractal noise is a)…procedural texture which spreads a colour across the surface in a chaotic manner"*
> - Bill Fleming, 3D Modelling & Surfacing, 1999

Figure 51 left shows the result of this with amplitude set to 1.0, threshold 0.0, ratio 0.707, frequency ratio 2.0, levelMin 0.0, levelMax 9.0 and default colour set to grey

Stage 3.

The surface is still a little too generic I feel and could do with further variation. Using a Brownian node, lunarity 4.0, increment 0.0, octaves 5, default colour grey, colourGain beige and colourOffset dark brown we get the more subtle surface texture shown below in figure 52. This more stippled effect, multiplied with the previous node connections, would give us the variety of surface densities we need (as shown in figure 53 also below).

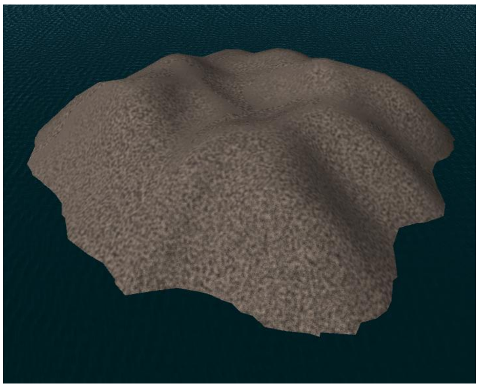


Figure 52 Stage 3a - Texturing the rock Surface

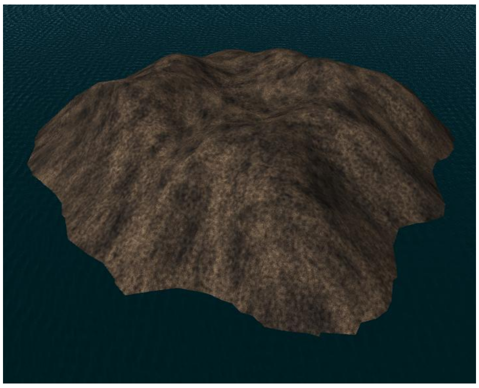


Figure 53 Stage 3b - Texturing the rock surface

Figure 54 Stage 4 - Texturing the rock surface

Stage 4 – Creating a bump map.

Starting with a Fractal node, amplitude 1.0, threshold 0.0, ratio 1.0, frequency ratio 1.0, levelMin 0.0, levelMax 25, default colour set to black, colourGain set to a dark beige and colourOffset set to brown gives a wavy texture simulating the ripples sometimes found in rocks as they build up over the years.

Stage 5.

Using another Fractal node, this time set with amplitude 1.0, threshold 0.0, ratio 1.0, frequency ratio 2.0, levelMin 0, levelMax 9, default colour dark grey, colourGain light grey and colourOffset green. I have chosen to use green in this node as I am planning to use it later on for Diffuse and ambientColour to simulate the presence of mould and weathering. As you can see below in figure 55, this bump is very detailed and evenly spread unlike the previous one which is more concentrated in certain areas. If I combine the two with a multiply node I get a more desired rock bump (as can be seen in figure 56, also below).

Figure 55 Stage 5a - Texturing the rock Surface

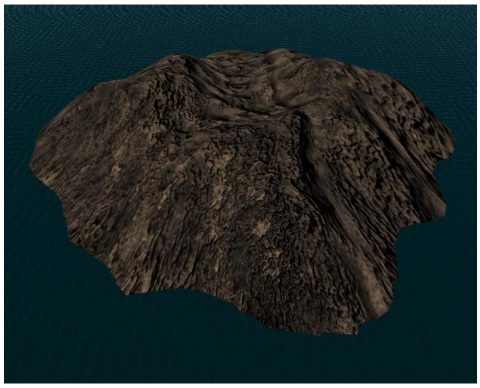


Figure 56 Stage 5b - Texturing the rock surface

Stage 6 – Creating diffuse.

As mentioned above, I am going to use the previously created green fractal for the diffuse. By adding this colour to the texture it suggests the weathering of the rock over time. All rocks are weathered by the local weather conditions and are particularly susceptible to moisture in the atmosphere which can cause moulding, erosion and the growth of mosses and grasses in the cracks of the surface.

Figure 57 Stage 6 - Texturing the rock surface

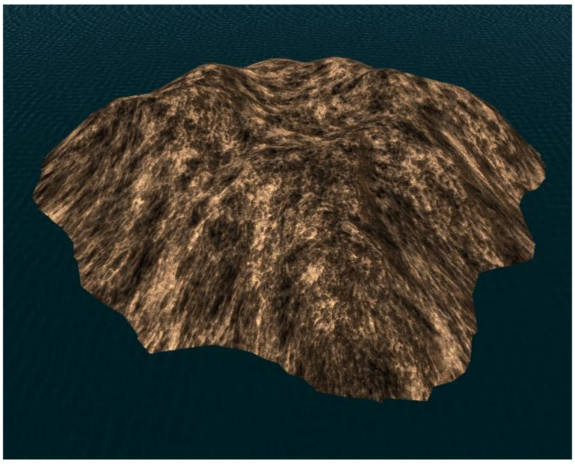
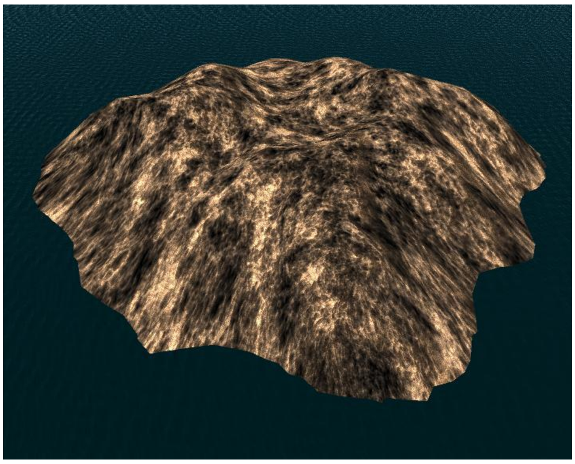


Figure 58 Stage 7 - Texturing the rock surface

Stage 7 – Creating ambient colour.

Going back to the original network of nodes created for the colour, using the output of the rock node before being passed through the fractal node, will give a colour specific distribution of the ambient colour. Looking at the figure 58 left, you can see that this means the rock with have highlights in areas of the colour texture where the colour itself is already brighter. This is good as it is quite realistic but still, some darker areas of the rock would in reality, also be lit too.



Stage 8.
By passing the outColour of this rock into the colourGain of the fractal as done before for the colourMap give an even more colour specific highlight distribution.

Figure 59 Stage 8 - Texturing the rock surface

Stage 9.
To vary these highlights slightly from the underneath colour layout they need to be affected by some independent nodes. Create a linearly interpolating circular ramp with yellow/orange/green colour values, setting the bias to yellow for warm sunlight. Turn the noise and noiseFrequency up to 1. This gives a nicely spread out scattering of colours across the surface suggesting an aging of the rock exterior and the presence of other materials growing on the surface.

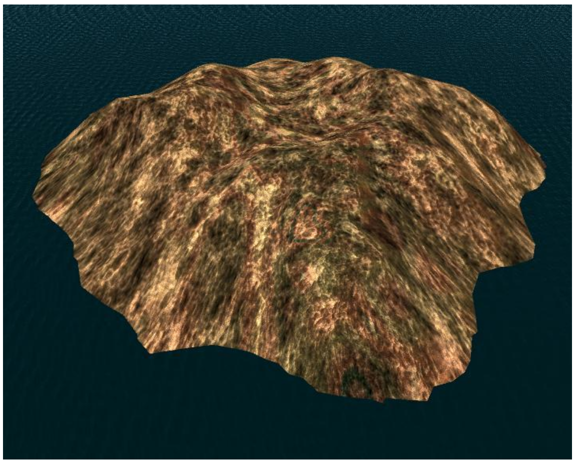


Figure 60 Stage 9 - Texturing the rock surface

Stage 10.
To combine the last two ambientColours I am going to do this through a marble node to increase the subtlety and variation even further. Plug the previously made ramp into the colour attribute of this new marble node. Connect the outValue from an Rgb-to-Luminance Utility node, used to convert the output of the previous fractal, into the veinWidth attribute of the marble. This means that the nice spread of colours we achieved with the ramp will show through as the colour in the marble pattern and the fractal will cause the veins in this colour to vary in a more natural manner. The final result can be seen below in figure 61.

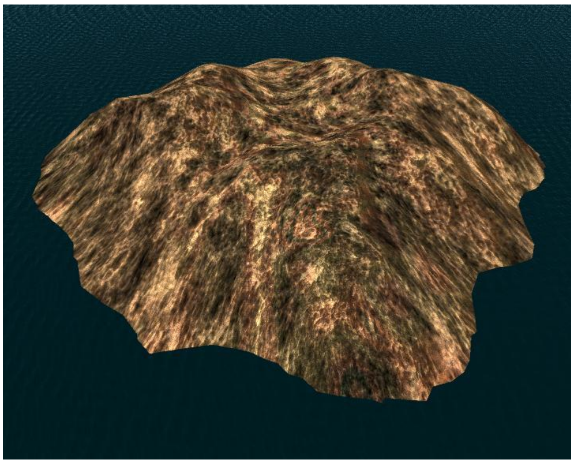


Figure 61 Stage 1 - Texturing the rock surface

The Final Images

Now, having researched the methods behind procedural texturing using the Maya Hypershade and completed a number of texturing exercises I felt ready to create my final images. I chose six images of different frog sub-species in different habitats. My aim was to recreate as close to reality as possible, the entire images using only procedural textures. The result of my work can be seen as follows.

Image One – Blue Azureus Poison Arrow Frog on Dried Leaves

Original Image:

Figure 62 Blue Poison Arrow Frog. Source: www.dartfrog.co.uk

Textured Image:

Figure 63 My textured version of the Blue Poison Arrow Frog

Shaders Used:

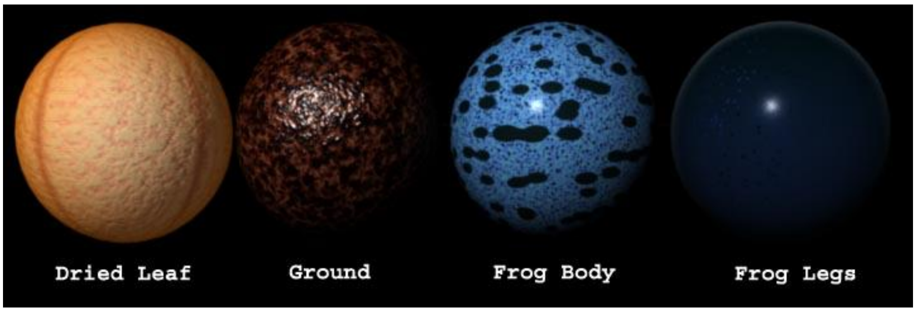


Figure 64 Examples of textures used in 'Image One'

Shading Networks:

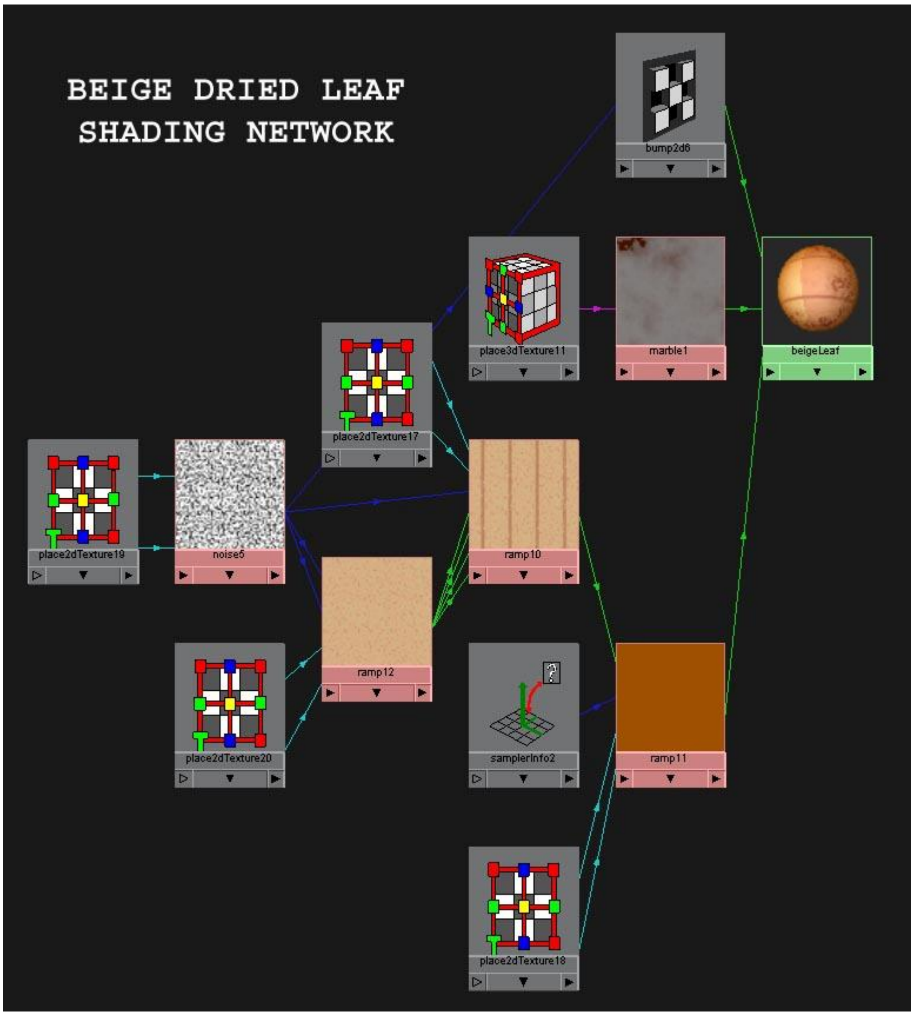


Figure 65 Shading Network for the beige dried leaf texture

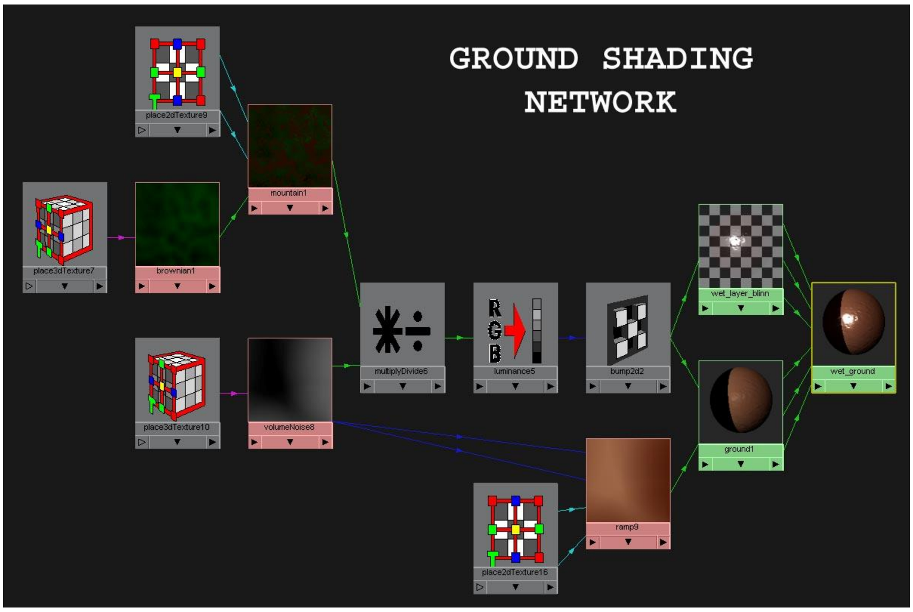


Figure 66 Shading network for the ground texture

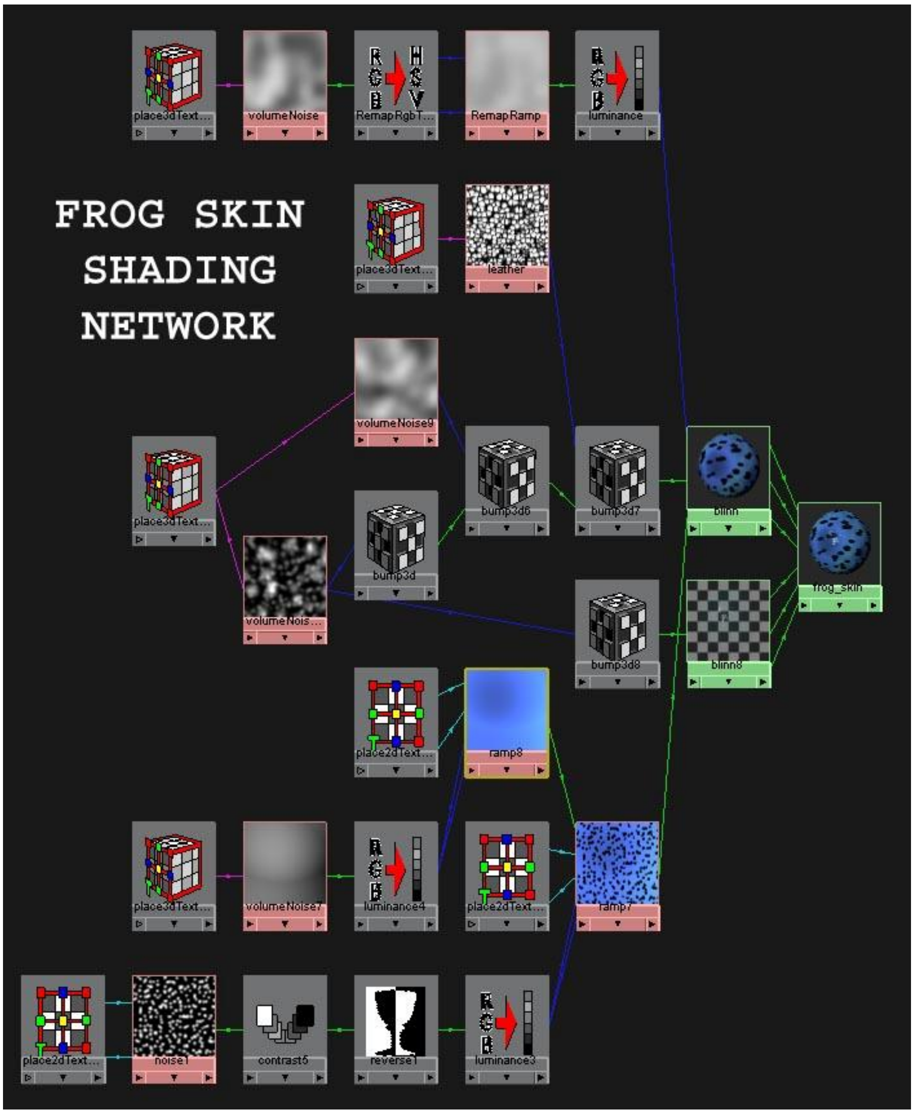


Figure 67 Shading network for the frog skin texture

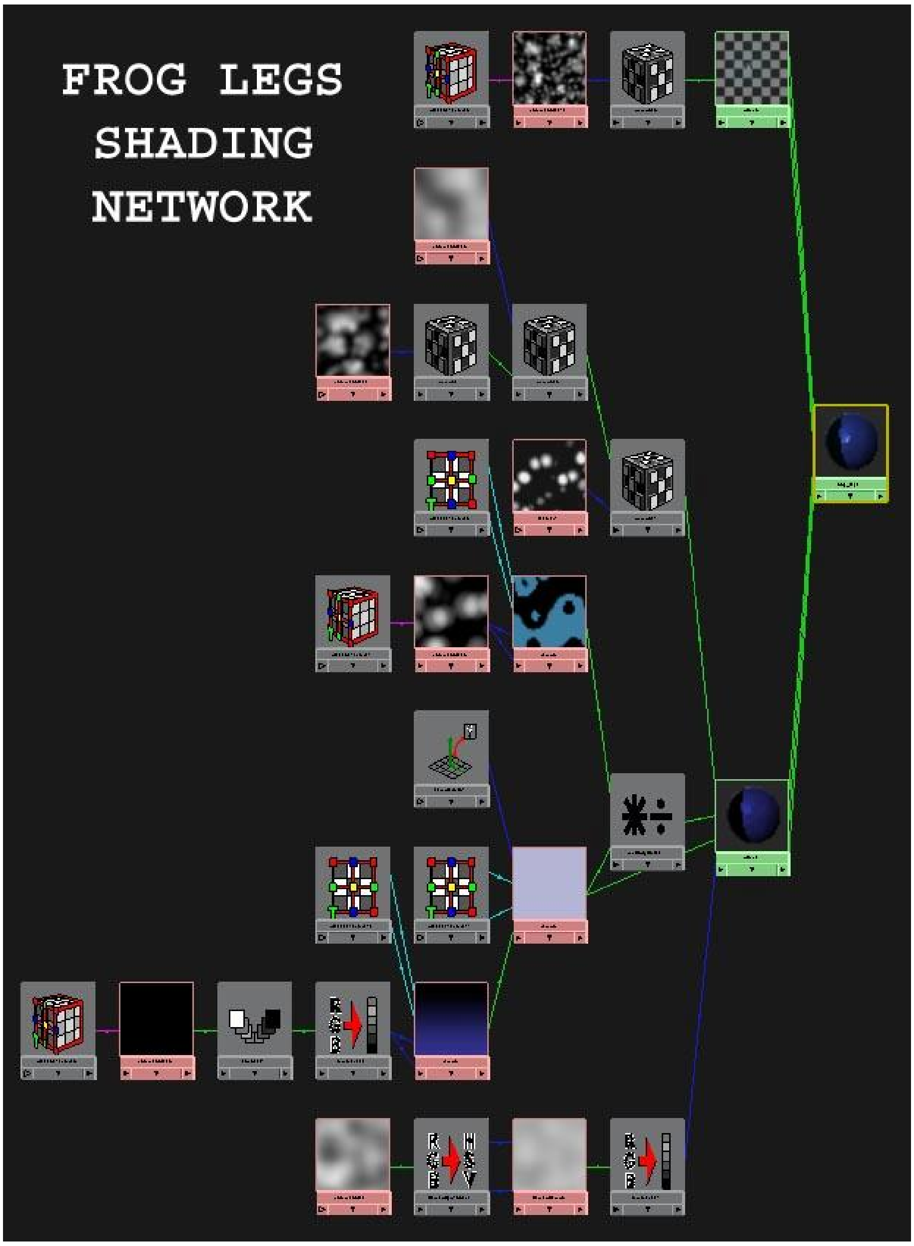


Figure 68 Shading network for the frog legs texture

Image Two – Yellow and Black Poison Arrow Frog on Tropical Leaves

Original Image:

Figure 69 Yellow and Black Poison Arrow Frog. Source: www.blackjungle.co.uk

Textured Image:

Figure 70 My textured version of the Yellow and Black Poison Arrow Frog

Shaders Used:

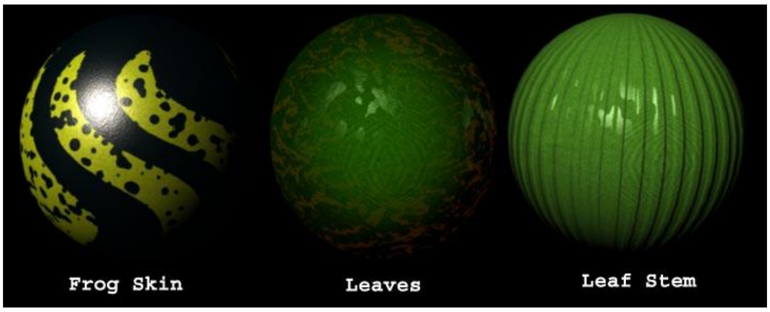


Figure 71 Example of textures used in ‘image two’

Shading Networks:

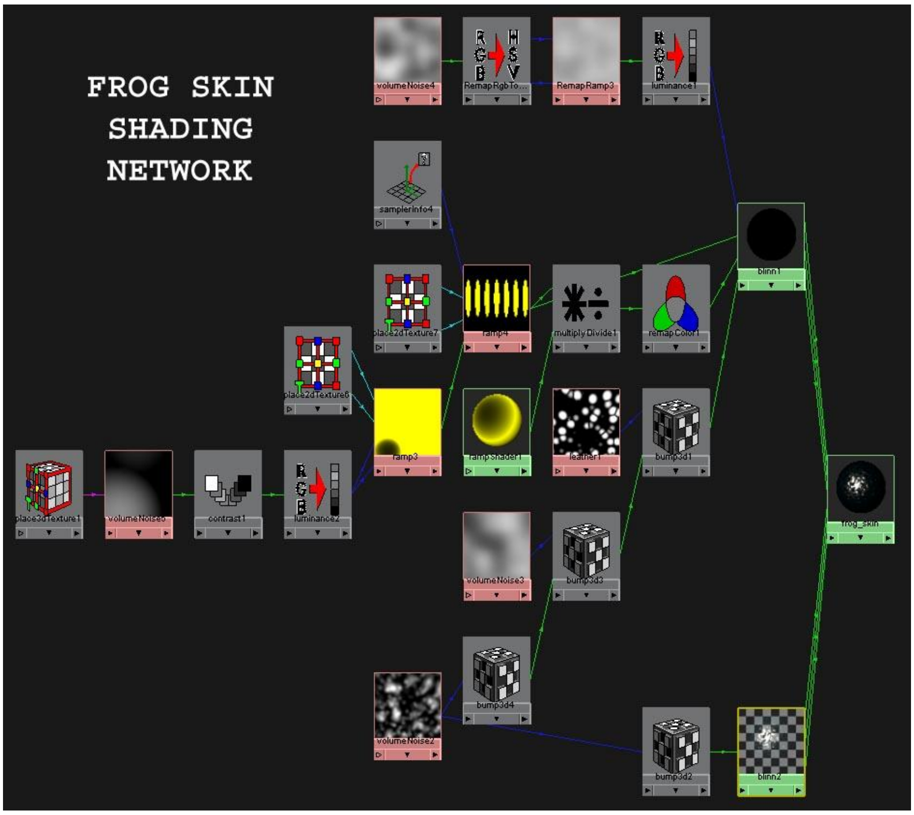


Figure 72 Shading network for frog skin texture

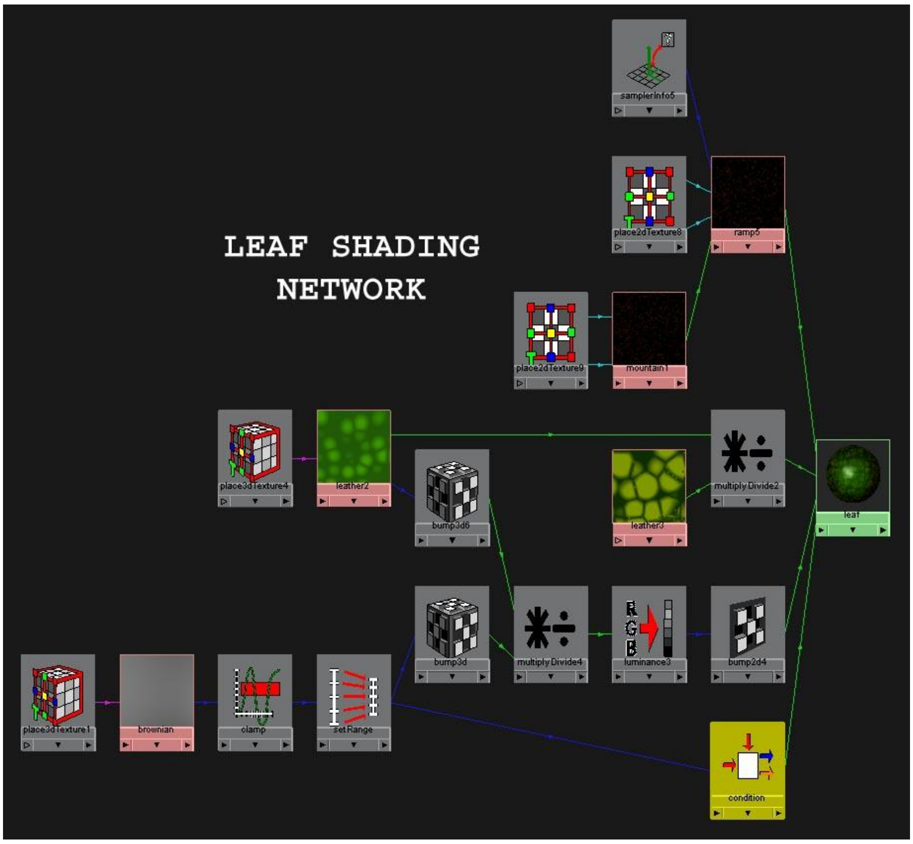


Figure 73 Shading Network for leaf texture

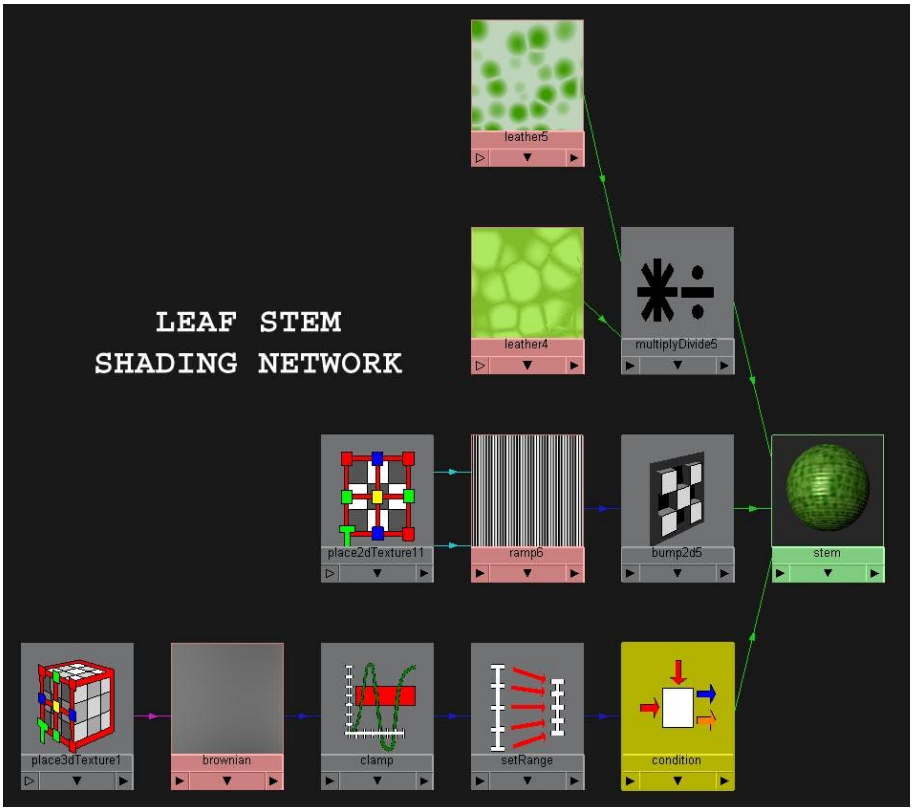


Figure 74 Shading network for leaf stem texture

Image Three – Dyeing Poison Dart Frog on Wet Ground

Original Image:

Figure 75 Dyeing Poison Dart Frog. Source: www.be.wednet.edu

Textured Image:

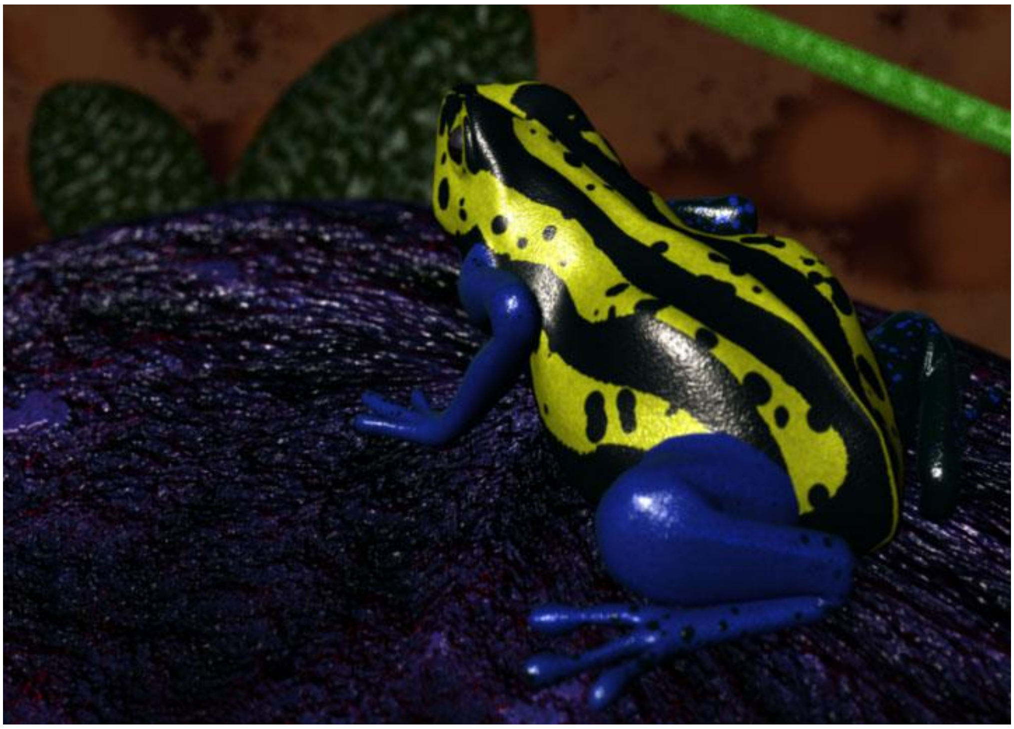


Figure 76 My textured version of the Dyeing Poison Dart Frog

Shaders Used:

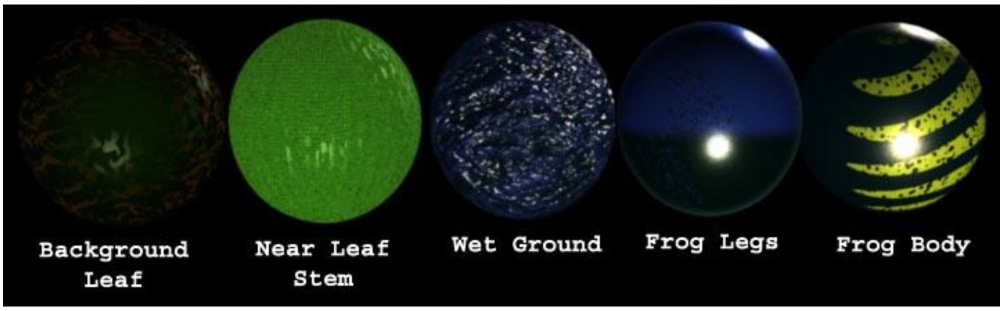


Figure 77 Example of shaders used in 'image three'

Shading Networks:

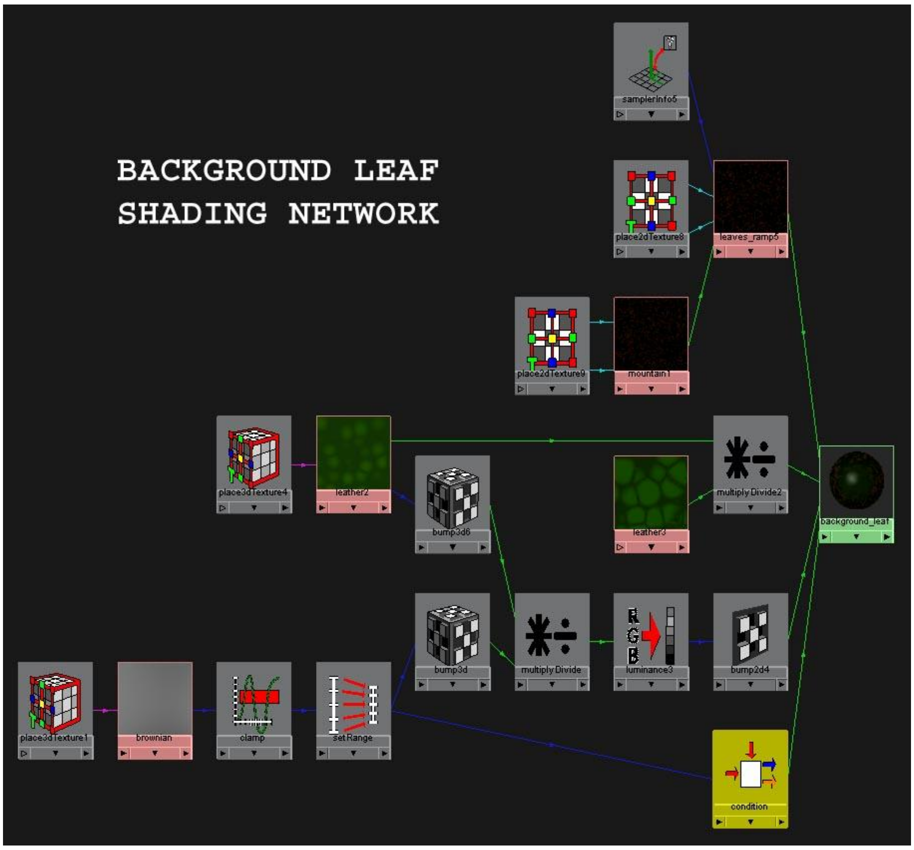


Figure 78 Shading network for background leaf texture

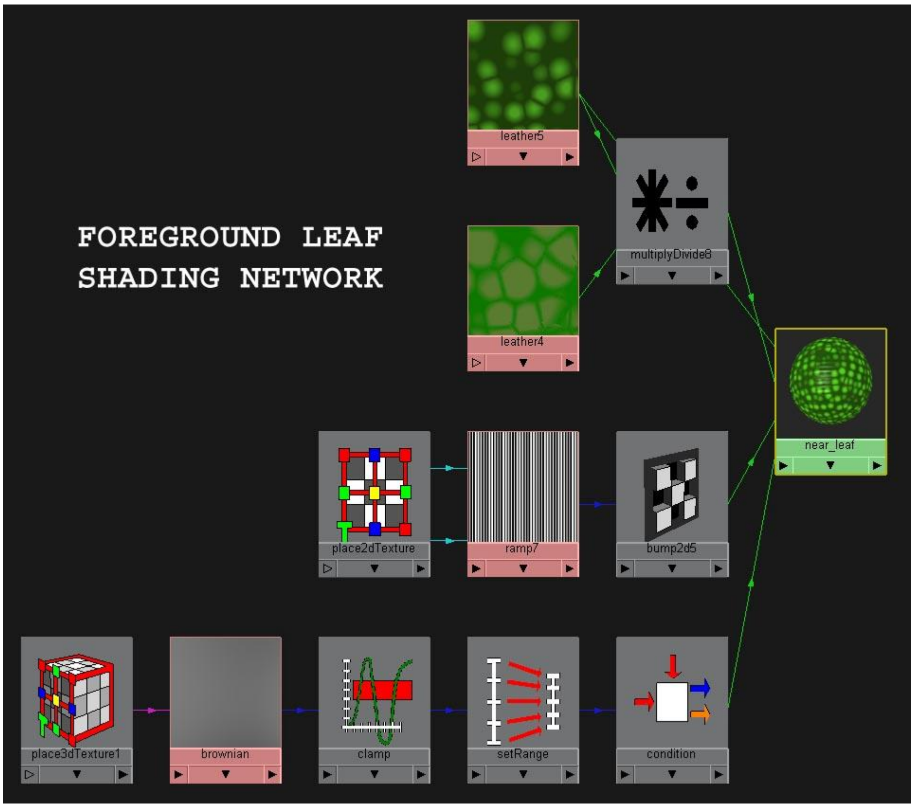


Figure 79 Shading network for foreground leaf texture

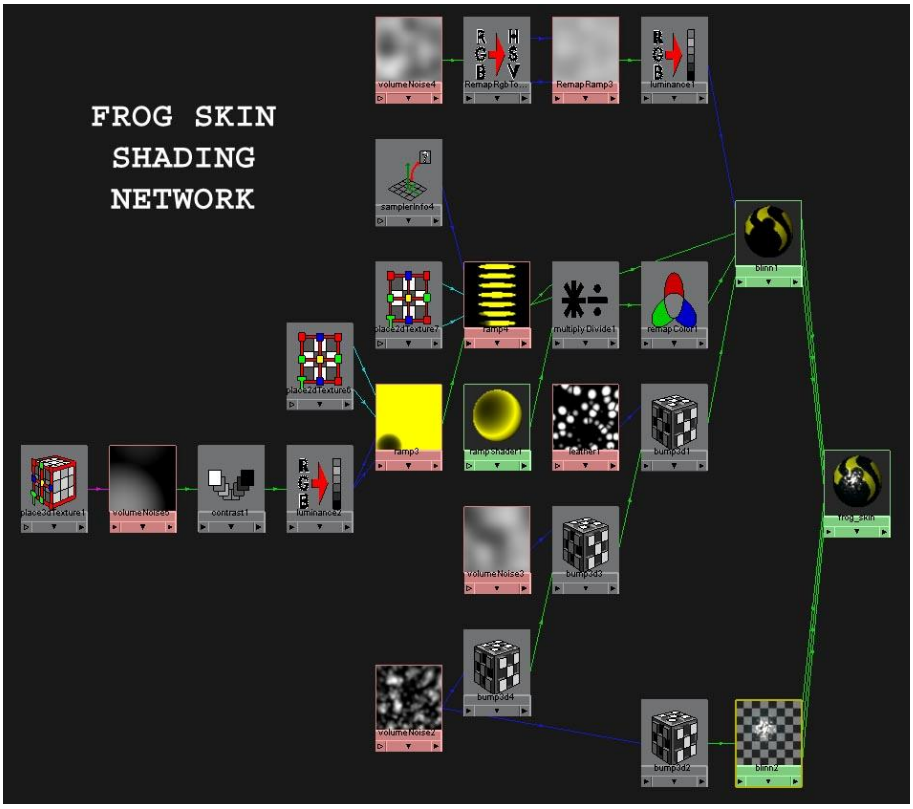


Figure 80 Shading network for frog skin texture

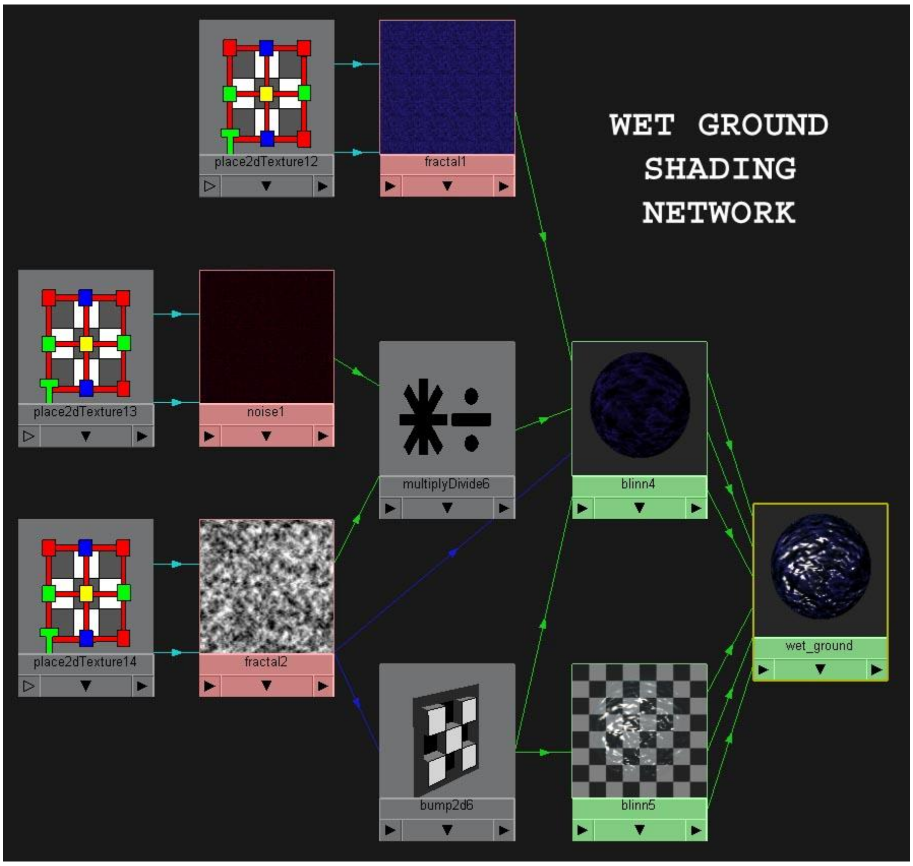


Figure 81 Shading network for wet ground texture

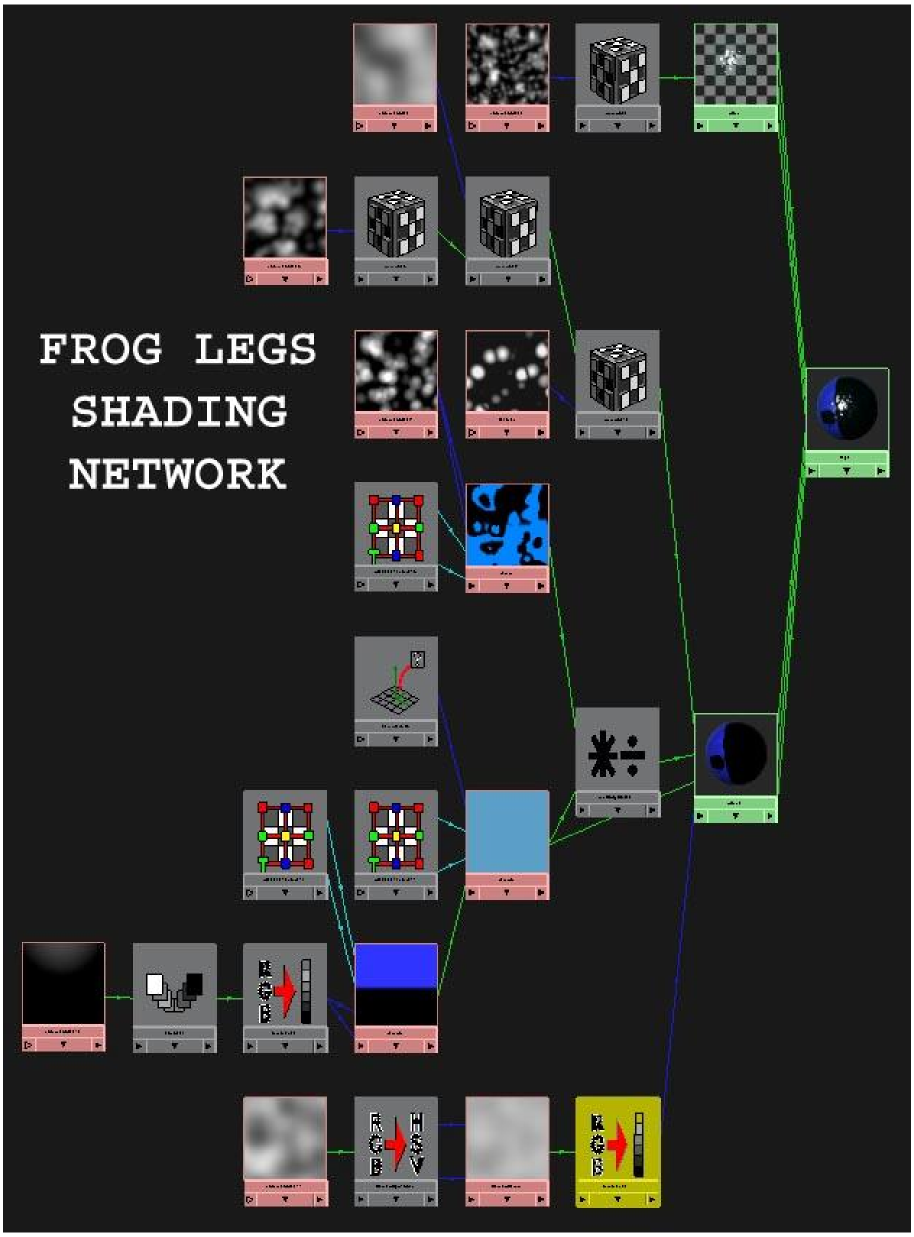


Figure 82 Shading network for frog legs texture

Image Four – Yellow and Black Poison Dart Frog in Puddle

Original Image:

Figure 83 Yellow and Black Poison Arrow Frog. Source: www.geodyssey.co.uk

Textured Image:

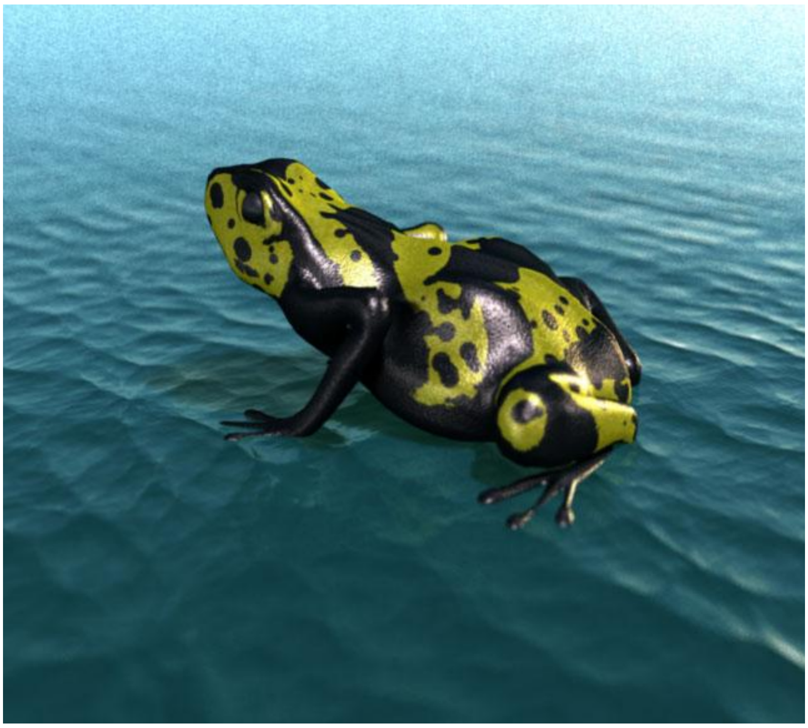


Figure 84 My textured version of the Yellow and Black Poison Arrow Frog.

Shaders Used:

Figure 85 Example of shaders used in ‘image four’

Shading Networks:

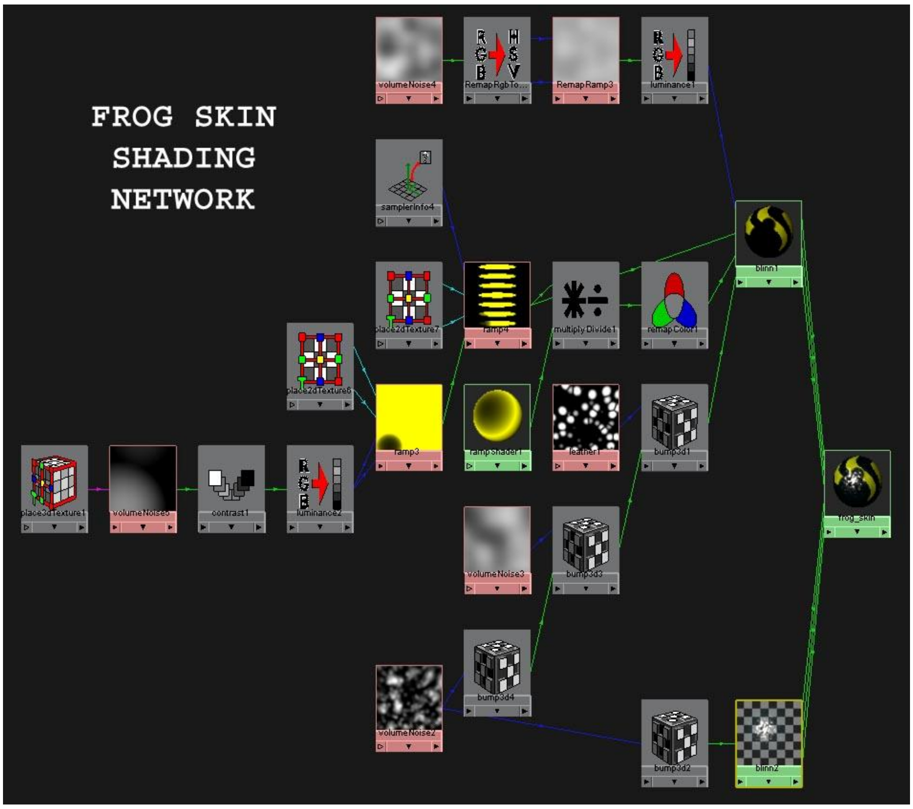


Figure 86 Shading network for frog skin texture

Image Five – Orange-Blue Poison Dart Frog on Leaf

Original Image:

Figure 87 Orange-Blue Poison Arrow Frog. Source: www.casarioblanco.com

Textured Image:

Figure 88 My textured version of the Orange-Blue Poison Arrow Frog

Shaders Used:

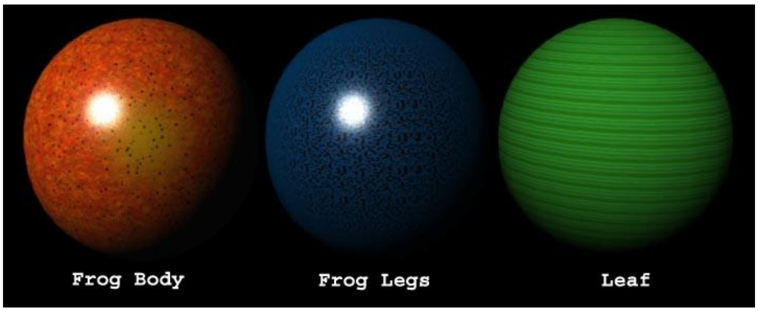


Figure 89 Example of shaders used in 'image five'

Shading Networks:

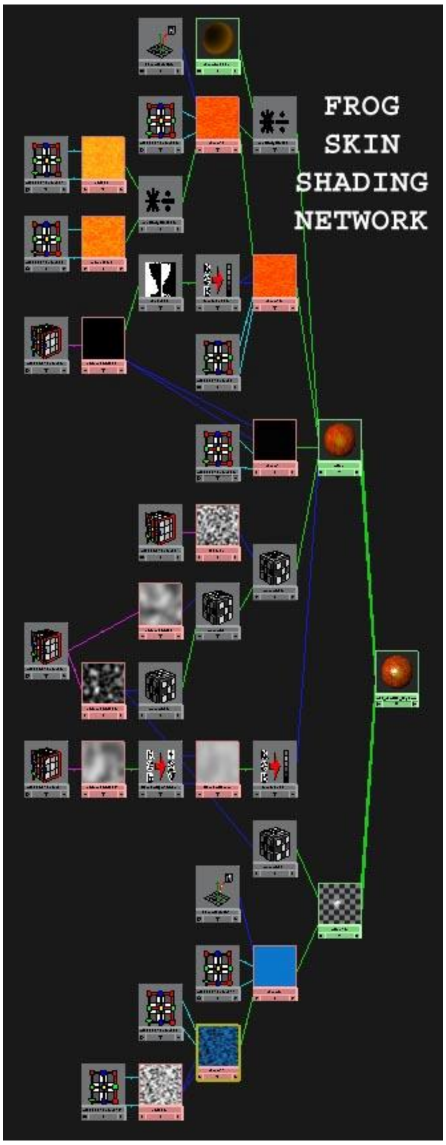


Figure 90 Shading network for frog skin texture

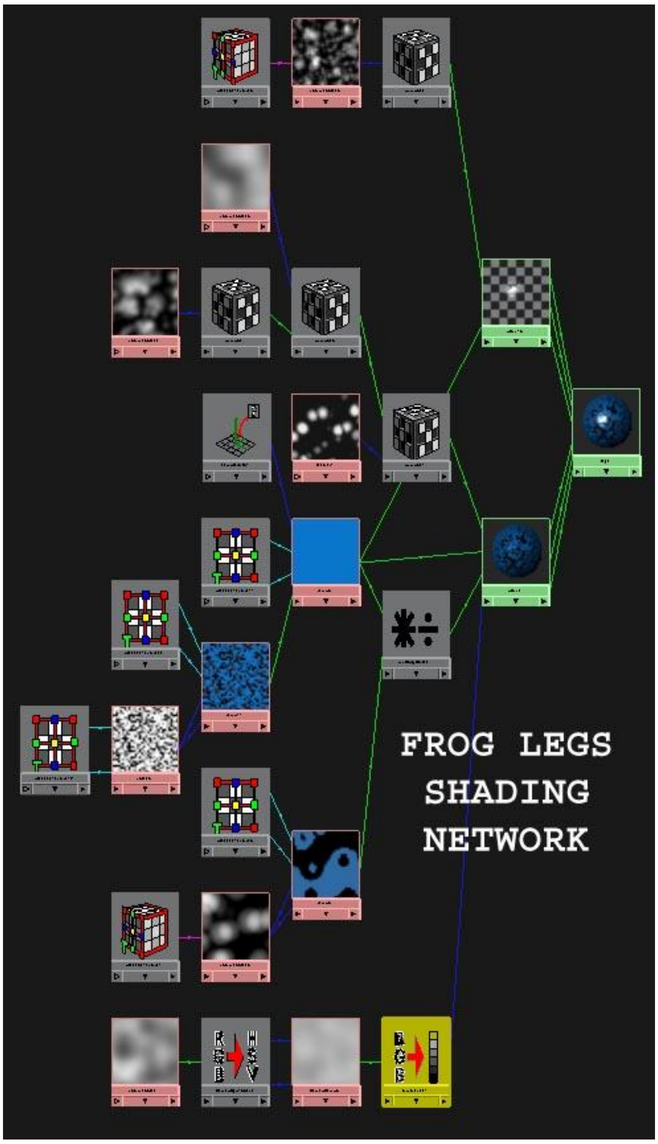


Figure 91 Shading network for frog legs texture

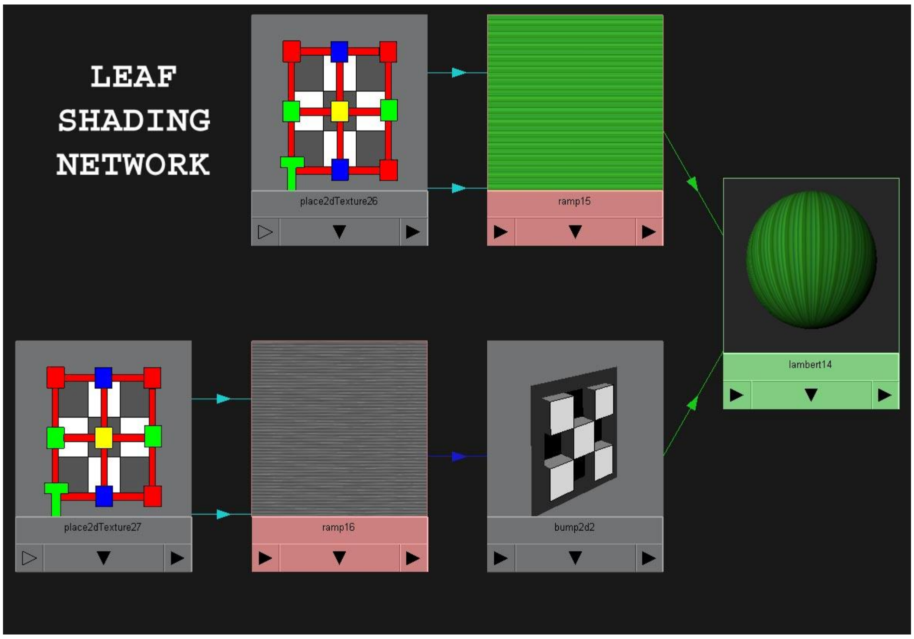


Figure 92 Shading network for leaf texture

Image Six – Red Poison Arrow Frog on Log

Original Image:

Figure 93 Red Poison Arrow Frog. Source: www.advantagepanama.com

Textured Image:

Figure 94 My textured version of the Red Poison Arrow Frog

Shaders Used:

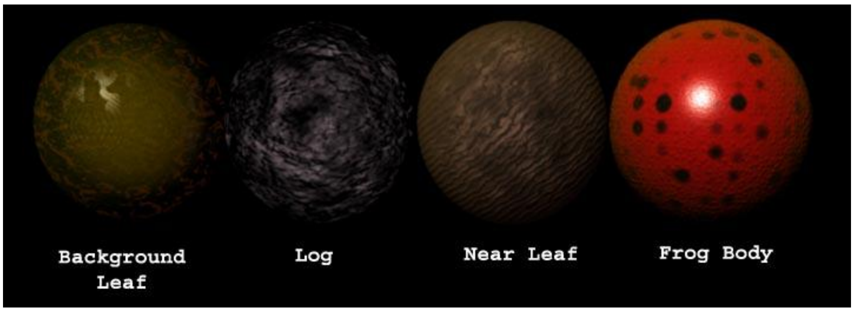


Figure 95 Example of shaders used in 'image six'

Shading Networks:

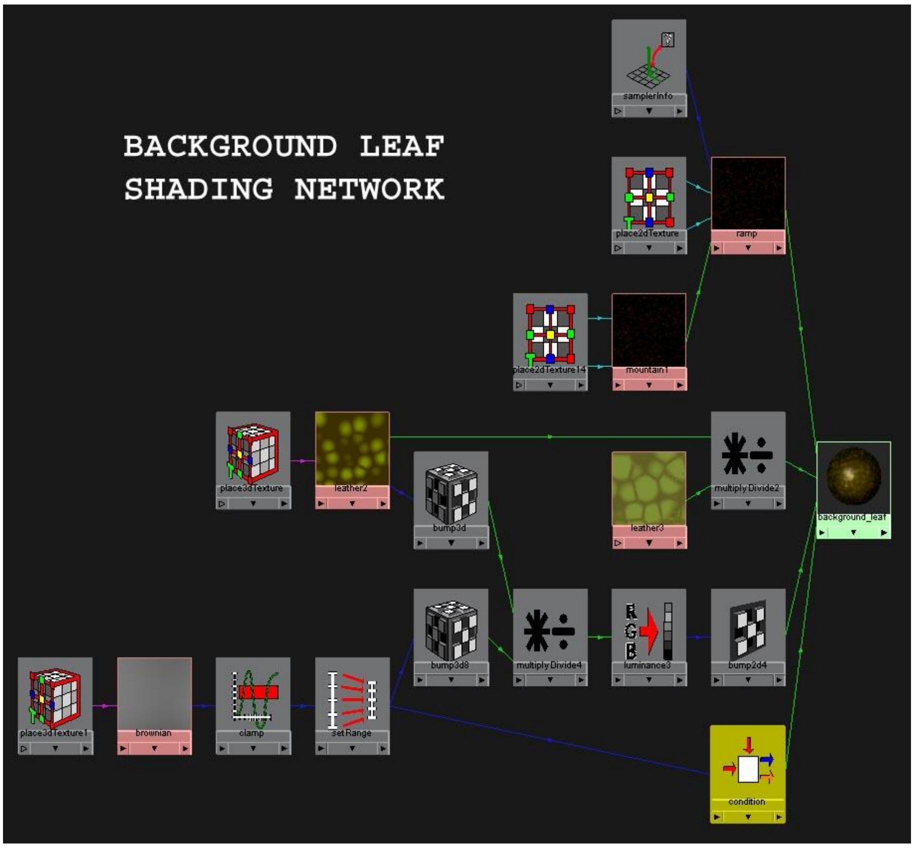


Figure 96 Shading network for background leaf texture

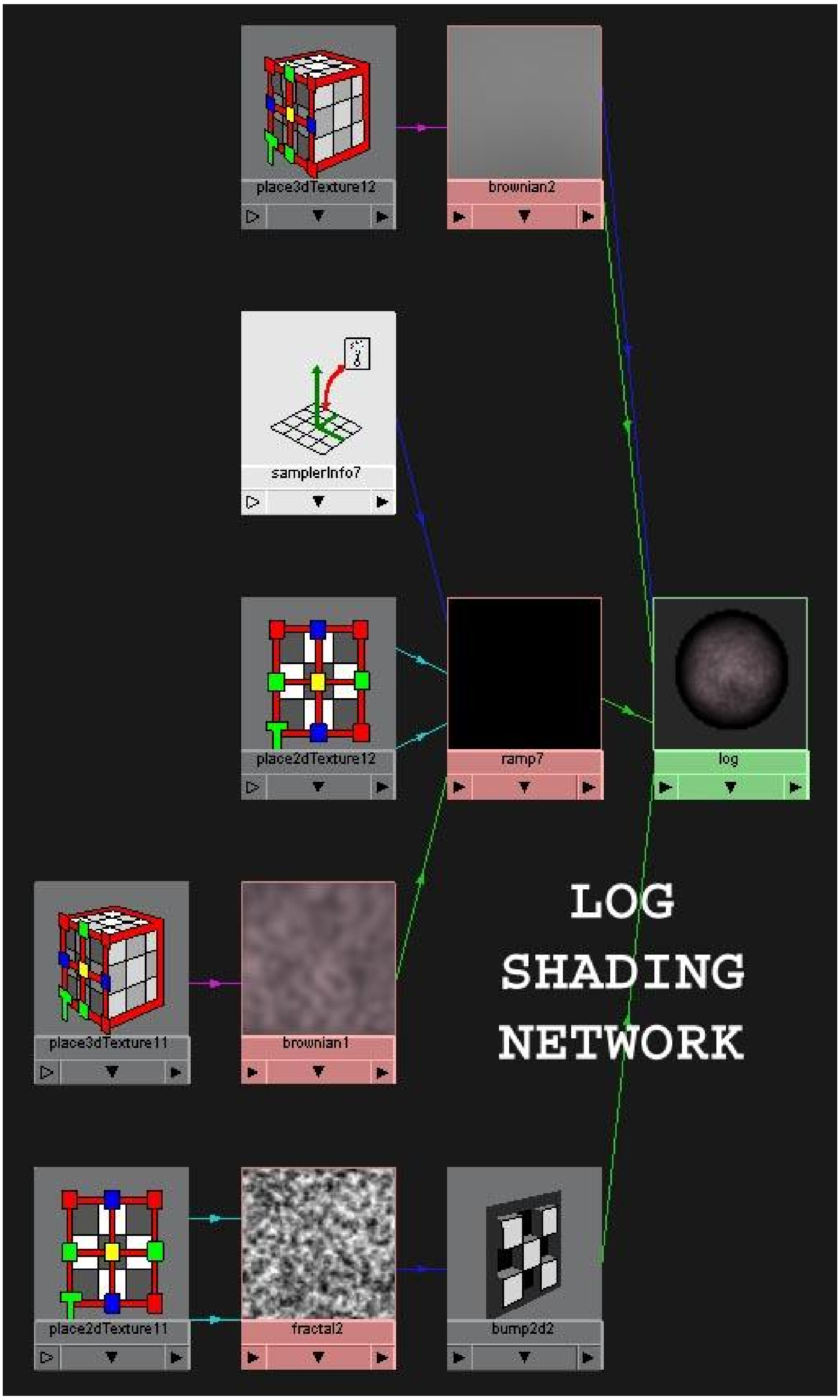


Figure 97 Shading Network for log texture

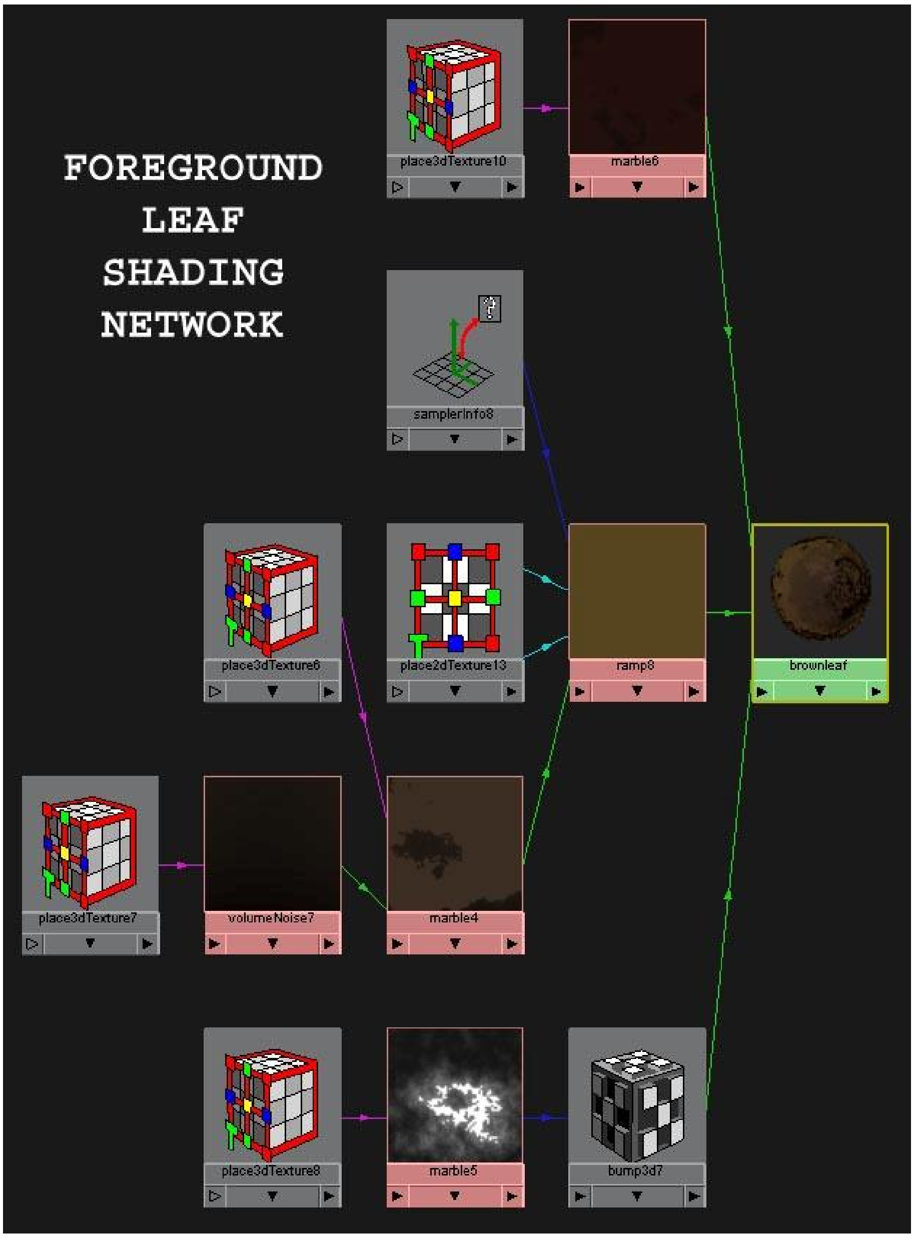


Figure 98 Shading network for foreground leaf texture

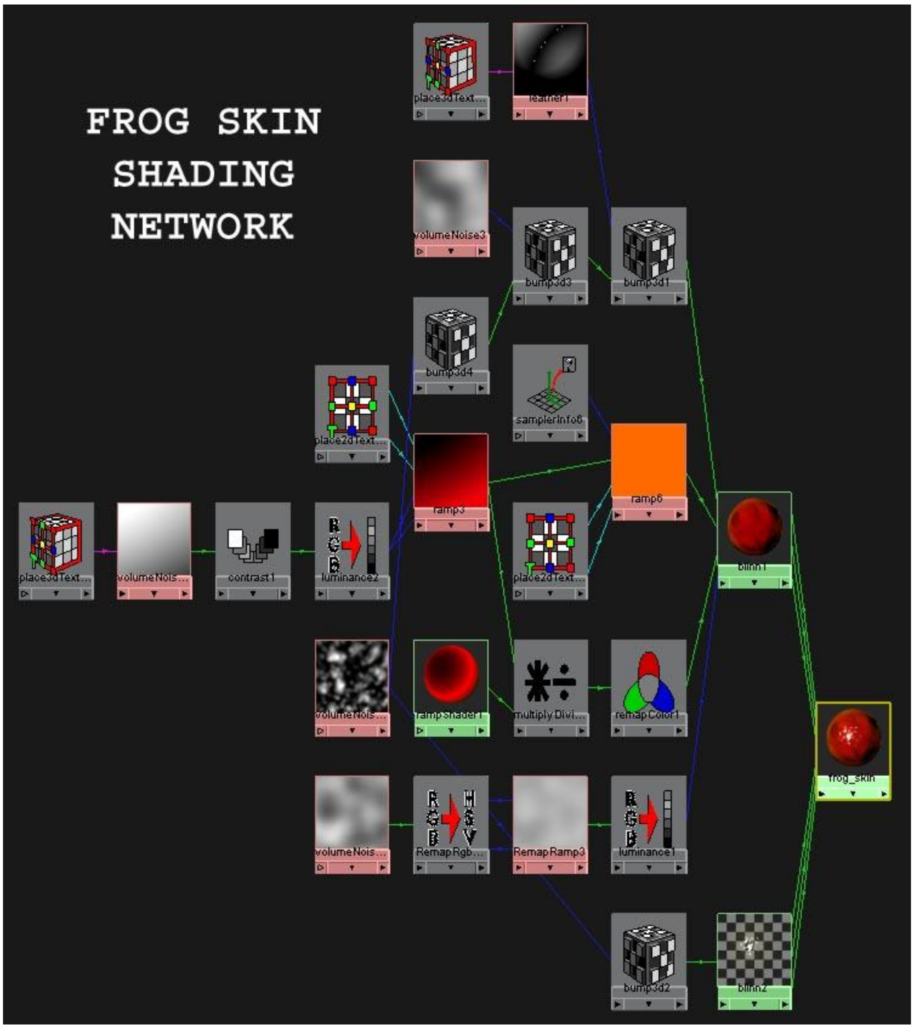


Figure 99 Shading network for frog skin texture

Conclusion

Procedural texturing is the method by which we use the computer to generate our texture maps using algorithms designed to create surfaces which simulate those in real life. Unlike with hand painted texture maps, procedural algorithms allow us to save on storage space for the scene, offer us infinite resolution, no seams and no need for mapping coordinates. As well as this, if you know your way around the Hypershade, they can be a very fast way of creating the surface you need. However, when it comes to rendering, the complex details from the scene or sequence are abstracted into a procedure which then evaluates as and when it is needed which can increase render times. Although procedural textures may be good to get results fast and relatively cheaply; they are not always the best way to go about texturing your scene.

When it comes to texturing a photorealistic scene the most important thing to do is research your surface. When doing so you will always notice that no two surfaces are ever the same, despite perhaps being made of the same material. This is because things in the real world interact with life constantly. A table will have scratches from where cutlery has been dropped and scraped over years of use; a pen will have grip marks where it has been held over and over again in the same place. Even new things will always show signs of interaction, perhaps a little dust, smeared fingerprints or the slightest tea stain. The fact is that every surface has a history to it, a use and therefore a unique fingerprint itself. There is no way that any procedural texture is ever going to be able to completely represent the past of an object like a hand painted texture could. To make something look real you have to understand where it has been and therefore how it is likely to be used in the future. Hand pained texturing is the only way to fully achieve this level of reality. However, it is important not to forget the impact of modelling in a scene also. According to Bill Fleming:

> *"There is much more to photorealism than applying real image textures to objects. The way you stage a scene defines the realism."*
>
> \- Bill Fleming, Advanced 3D Photorealism Techniques, 1999[10]

Unfortunately, in industry today, it is often the case that a nearing deadline will force texture artists to turn to procedural methods to save time. There are also a number of surfaces which would be impossible to create without writing a procedure to calculate it. For example, water or any other liquid surface would require an algorithm. Large or organic objects are particularly good for procedural texturing as they give a completely random and seamless effect. Also, any surface with a relatively uniform texture need not be textured by hand. Therefore, it is fair to say that both methods have their place in the computer animation pipeline. Knowledge of both processes is essential for the modern texture artist.

This innovations report has been invaluable for me personally. When I began I had little prior knowledge of the Hypershade let alone the algorithms involved in

[10] Advanced 3D Photorealism Techniques, Copyright Academic Press, 1999

creating each node. I have learnt a great deal of relevant information about this topic which I intend to put to good use in the future. Although I have concluded that there is no substitute for a detailed hand painted texture map when it comes to photorealistic scenery, I appreciate the valuable place the procedures have in it also. They each depend heavily on each other and carrying out this assignment has really made me realise this. I now believe an understanding of the maths behind the art is essential. Despite how much I have already learnt, there is still a great deal of things to explore in this area which I intend to continue with.

Figure 100 Rendered shot of all 6 final frogs together

References

ü Books

• Texturing & Modelling	D. S. Ebert F. K. Musgrave D. Peachey K. Perlin S. Worley	Academic Press, 1998
• Advanced 3D Photorealism Techniques	B. Fleming	John Wiley & Sons, Inc, 1999
• 3D Modelling and Surfacing	B. Fleming	Academic Press, 1999
• Learning Maya 6 – Rendering	C. Berndt P. Gheorghian J. Harrington A. Harris C. McGinnis	Alias Systems, 2004
• Digital Texturing & Painting	O. Demers	New Riders Publishing, 2002
• Using Maya – Rendering	P. Anderson S. Brooks J. Dila L. Ford C. Hunter A. Kozyniak C. Macri J. MacPhail S. Park D. Ramey L. Rose M. Stivers	Alias/Wavefront, 1999
• 3D Photorealism Toolkit	B. Fleming	John Wiley & Sons, Inc, 1998

ü Websites

Frog Information and Images

- www.accefyn.org.co/ranas/frogs/about_frogs.htm
- www.aqua.org/animals_bluepoisondartfrog.html
- www.arkive.org/species/GES/amphibians/Micrixalus_nudis/
- www.beachhouselinens.com/catalog/index.php/cPath/19_72_234
- www.belowwater.com/
- www.casarioblanco.com/poison-dart-frog.html
- www.dartfrog.co.uk/amphibians.html
- www.en.wikipedia.org/wiki/Poison_arrow_frog
- www.en.wikipedia.org/wiki/Poison_dart_frog
- www.enchantedlearning.com/subjects/amphibians/Poisonfrogstrawbprintout.shtml
- www.exploratorium.edu/frogs/mainstory/index.html
- www.greglasley.net/toadsandfrogs
- www.lsb.syr.edu/projects/cyberzoo/poisonarrowfrog.html
- www.nashvillezoo.org/blfrog.htm
- www.nationalzoo.si.edu/Animals/Amazonia/Facts/fact-poisondartfrog.cfm
- www.oregonzoo.org/Cards/Amazon/blue.poison.dart.frog.htm
- www.poweredtemplates.com/search/tropical%20frog/6/1/index.html
- www.rainforest-australia.com/frogs.htm
- www.wildherps.com/families/Leptodactylidae.html

Texturing Information and Examples

- www.3dlinks.com/shop/Details.cfm?ProductID=667
- www.3dmd.net/MAYA_texturing_tutorials.htm
- www.3dtotal.com/team/Tutorials/leafproject/leaf_1.asp
- www.deathfall.com/article.php?sid=5697
- www.en.wikipedia.org/wiki/Blinn
- www.en.wikipedia.org/wiki/Brownian_motion
- www.en.wikipedia.org/wiki/Incandescence
- www.en.wikipedia.org/wiki/Phong
- www.en.wikipedia.org/wiki/Procedural_texturing
- www.en.wikipedia.org/wiki/Ramp
- www.foyles.co.uk/foyles/display.asp?ISB=9780471794042&TAG=&CID=
- www.highend3d.com/maya/downloads/shaders/page4.html?numi=334
- www.highend3d.com/maya/tutorials/texturing/193.html
- www.trickwerk.com/artic/xfrog2rib.php?lang=de
- www.tutorialized.com/tutorial/How-to-create-a-Wet-Shader/4077
- www.tutorialized.com/tutorials/Maya/Textures-and-Materials/1

Appendix

Photographs taken at Marwell Zoo, Dorset: