



FX SL Package

for Maya

User's Manual

Release 1.0
August 2005

CgFX Shading Language Package

ALL ASDN DESIGN SPECIFICATIONS, REFERENCE BOARDS, FILES, DRAWINGS, DIAGNOSTICS, LISTS, AND OTHER DOCUMENTS (TOGETHER AND SEPARATELY, "MATERIALS") ARE BEING PROVIDED "AS IS." ASDN MAKES NO WARRANTIES, EXPRESSED, IMPLIED, STATUTORY, OR OTHERWISE WITH RESPECT TO THE MATERIALS, AND EXPRESSLY DISCLAIMS ALL IMPLIED WARRANTIES OF NONINFRINGEMENT, MERCHANTABILITY, AND FITNESS FOR A PARTICULAR PURPOSE.

Information furnished is believed to be accurate and reliable. However, ASDN Web site assumes no responsibility for the consequences of use of such information or for any infringement of patents or other rights of third parties that may result from its use. No license is granted by implication or otherwise under any patent or patent rights of ASDN Web site. Specifications mentioned in this publication are subject to change without notice. This publication supersedes and replaces all information previously supplied. ASDN Web site products are not authorized for use as critical components in life support devices or systems without express written approval of ASDN Web site.

Trademarks

ASDN and the ASDN logo are trademarks of ASDN Web site.

Microsoft, Windows, the Windows logo, and DirectX are registered trademarks of Microsoft Corporation.

OpenGL is a trademark of SGI.

Maya is a trademark of Alias System.

NVIDIA is a trademark of NVIDIA Corporation

Mentalray is a trademark of Mental images.

Other company and product names may be trademarks of the respective companies with which they are associated.

Updates

Any changes, additions, or corrections will be posted at the ASDN Web site:

<http://www.asdn.net.cn/>

Refer to this site often to keep up on the latest changes and additions to the Cg language.

Copyright

Copyright ASDN 2004,2005



ASDN

www.asdn.net.cn



CgFX Shading Language Package

Introduction

The CgFx Shading Language Package contains 11 CgFX Shader scripts, which can be applied on different illumination models and realization of various material.

This set of CgFX scripts is based on CgFX 1.4 compiler and Maya7.0 CgFX Shader. The coordinate system match to the default Maya coordinate system.

Notice: If you want refer sample CgFX files in Maya7.0 Preset folder, the some bug you need to know. The big one is the calculation of space transform matrix from the example script in Maya 7.0 Preset is incorrect (from object space to world space, Rotating the object except for along the Z axis will lead to the miscalculation and display distortion).

How to use

Check the CgFX shader plugin in Maya 7.0's Plug-in Manager, create a new CgFX Shader in HyperShader. Import the script, link the map and position of lights (if necessary, click the right mouse button at the slot of input light position to choose the light you want to link with).

Notice: Polygon object should meet the requirement of CgFX Shader plugin, for Example, object scale should be no less than 1. Freezing the object before hand is recommended. In order to use the shader mesh data properly, you must know in which of the arrays the shader is expecting what data.

CgFX shader can only use one UVSet, click the right mouse button at TexCoord0 under Shader Mesh Data to choose the UVSet to be applied. CgFX shader defaultly uses object's default UV setting.

To ensure the correct calculation, altering the data of Shader Mesh Data is not recommended.

[Please read Maya Help Document for relevant operation.](#)

Efficiency

All scripts have been optimized for higher performance, but process speed will slow down when Shader support more effect. Recommend you select the befitting Shading Script for effect.

The efficiency of the scripts for Normal are higher than those for Bump, and have better controlling effect. Therefore For Normal is recommended. Bump maps can be converted to normal map through the [Photoshop Filter plugin](#) from NVIDIA.

Limitations

Limit come from Maya7.0 CgFX shader (The detail for Maya7.0 Help Document):

You cannot use CgFX shaders that contain certain effects in Maya:

- 1, Global effects
- 2, Multi-pass effects

So CgFX shader interact with environmental is limited, and can not be make global effect, for example: Glow, Highlight, Motion Blur etc. Because these limit, It can't compute all light in scene automatic. So CgFX shader need a light position input for lighting. The shader just compute one light in scene for general case.

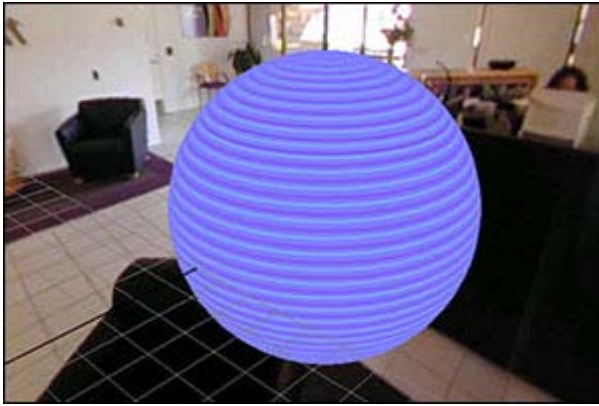
CgFX Shaders detail

The Shader Package includes 3 parts:

1. **Env material:** Environment material, for the environmental Cube map effect in the scene.
2. **For Normal map material:** Surface material, for the Normal map bump.
3. **For Bump map material:** Surface material, for the Bump map(High map) bump.

Common Technique

Often, shaders have more than one mode; for example, they may have modes to support two different video card or illumination models. You can adjust the technique in the Attribute Editor. All of shading scripts have two common effect in Technique (besides Env shader): [TangentNormal](#) and [ObjectNormal](#). When you select TangentNormal, Shader will display surface normal of tangent space. If you select ObjectNormal, Shader will convert tangent normal to object space and display.



TangentNormal



ObjectNormal

Env Material



cgEnv_M.cgfx

CG Artist will easy to view and comparison in scene,if display environmental(Cube map). But CgFX shader can't automatic display environmental by scene, because limit by self.

I added this shading script as new functionality of environmental effect.

Create a Sphere or Box(any shape) for the use of scene's environmental.

In the scene view, assign the CgFX material to your environmental object.

Connect this cgEnv_M.cgfx file to the CgFX Shader material in the Attribute Editor.

And you can see simulation environmental(based on Cube map) with perspective around your

scene (This shader can not be rendered in Maya Hardware render).

About type of Cube map please refer sample file (reflection.dds).

Technique:

EnvM: Direct project Cube map.

PerspectEnvM: Perspective project Cube map.

Parameter:

brightness: Controls the display brightness of Cube map.

contrast: Controls the display contrast of Cube map.

gamma: Controls the display gamma of Cube map.

envMapSampler: Input Cube map here.

For Normal Map Material



cgBlinn_M.cgfx

This shading script based on Blinn illumination model(for Game production).

Include effect: Diffuse, Specular, Reflection, Bump.

Technique:

BlinnM: Standard Blinn illumination model.

TangentNormal: Display surface normal of Tangent space.

ObjectNormal: Display surface normal of Object space.

Parameter:

bumpHeight: Controls how high the bumps display on the surface.

glossy: Controls the smoothness of surface for reflect Cube map.

specular: Controls the intensity of shiny highlights on the surface.

specExpon: Controls the size of shiny highlights on the surface.

lightPos: Input Light position here.

lightColor: Controls Light color and intensity.

ambiColor: Controls Ambient color of surface, Set to black by default.

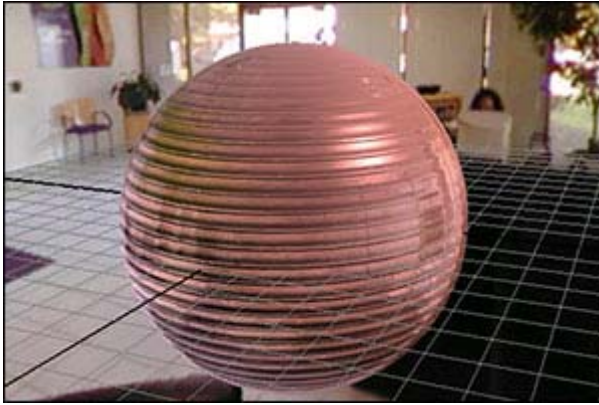
colorMapSampler: Input Diffuse color map here.

normalMapSampler: Input Normal map here.

glossyMapSampler: Input Glossy map here.

specMapSampler: Input Specular map here.

envMapSampler: Input Cube map here.



[cgAdBlinn_M.cgfx](#)

This advanced shading script based on Blinn illumination model.

Include effect: Diffuse, Specular, Reflection, Refraction, Fresnel, Chromatic Aberration, Bump, Backlighting, Simple subsurface scattering.

Technique:

AdBlinnM: Standard AdBlinn illumination model.

TangentNormal: Display surface normal of Tangent space.

ObjectNormal: Display surface normal of Object space.

Parameter:

bumpHeight: Controls how high the bumps display on the surface.

glossy: Controls the smoothness of surface for reflect Cube map.

specular: Controls the intensity of shiny highlights on the surface.

specExpon: Controls the size of shiny highlights on the surface.

transparency: Controls the transparency of object.

refractionIndex: Controls the Refraction Index of different wavelengths(Chromatic Aberration).

sssScale: Controls the intensity of Subsurface scattering.

backlightScale: Controls the intensity of backlighting.

rollOff: Controls the intensity of falloff for Subsurface scattering.

refMin: Controls the min value of reflection for Fresnel.

refMax: Controls the max value of reflection for Fresnel.

fresnelRatio: Controls the intensity of Fresnel. This control will effect transparency.

diffuseScale: Controls the intensity and range of Diffuse.

lightPos: Input Light position here.

lightColor: Controls Light color and intensity.

ambiColor: Controls Ambient color of surface, Set to black by default.

colorMapSampler: Input Diffuse color map here.

normalMapSampler: Input Normal map here.

glossyMapSampler: Input Glossy map here.

specMapSampler: Input Specular map here.

sssMapSampler: Input Subsurface scattering map here.

backlightMapSampler: Input Backlighting map here.

envMapSampler: Input Cube map here.



[cgMetal_M.cgfx](#)

This shader based on special illumination model for particular effect at simulating metallic surfaces.

Include effect: Diffuse, Specular, Reflection, Bump.

Technique:

MetalM: Standard Metal illumination model.

TangentNormal: Display surface normal of Tangent space.

ObjectNormal: Display surface normal of Object space.

Parameter:

bumpHeight: Controls how high the bumps display on the surface.

glossy: Controls the smoothness of surface for reflect Cube map.

specular: Controls the intensity of shiny highlights on the surface.

specExpon: Controls the size of shiny highlights on the surface.

dirtDiffuse: Controls the intensity and range of Diffuse.

lightPos: Input Light position here.

lightColor: Controls Light color and intensity.

ambiColor: Controls Ambient color of surface, Set to black by default.

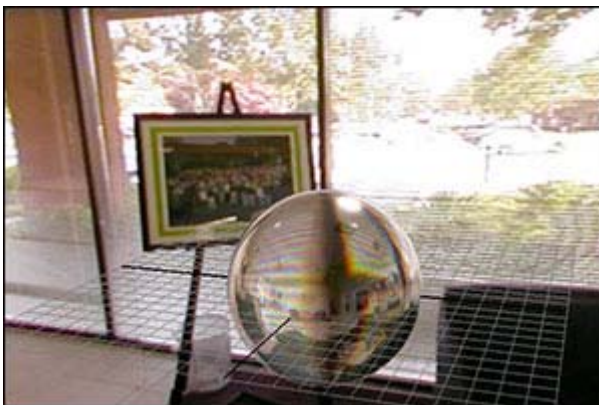
colorMapSampler: Input Diffuse color map here.

normalMapSampler: Input Normal map here.

glossyMapSampler: Input Glossy map here.

specMapSampler: Input Specular map here.

envMapSampler: Input Cube map here.



[cgGlass_M.cgfx](#)

This glass shader based on Blinn illumination model.

Include effect: Diffuse, Specular, Reflection, Refraction, Fresnel, Chromatic Aberration, Bump.

Technique:

GlassM: Standard Glass illumination model.

TangentNormal: Display surface normal of Tangent space.

ObjectNormal: Display surface normal of Object space.

Parameter:

bumpHeight: Controls how high the bumps display on the surface.

glossy: Controls the smoothness of surface for reflect Cube map.

specular: Controls the intensity of shiny highlights on the surface.

specExpon: Controls the size of shiny highlights on the surface.

transparency: Controls the transparency of object.

refractionIndex: Controls the Refraction Index of different wavelengths(Chromatic Aberration).

refMin: Controls the min value of reflection for Fresnel.

refMax: Controls the max value of reflection for Fresnel.

fresnelRatio: Controls the intensity of Fresnel. This control will effect transparency.

diffuseScale: Controls the intensity and range of Diffuse.

lightPos: Input Light position here.

lightColor: Controls Light color and intensity.

ambiColor: Controls Ambient color of surface, Set to black by default.

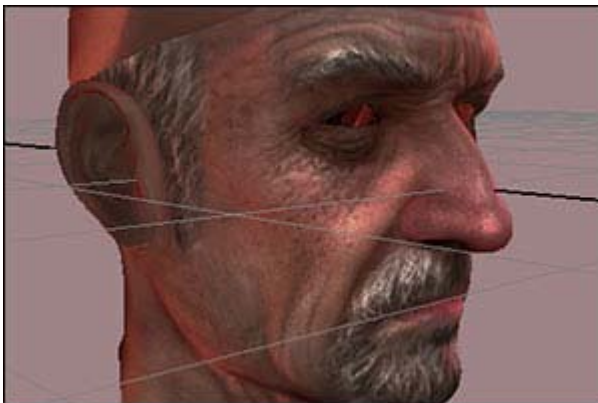
colorMapSampler: Input Diffuse color map here.

normalMapSampler: Input Normal map here.

glossyMapSampler: Input Glossy map here.

specMapSampler: Input Specular map here.

envMapSampler: Input Cube map here.



cgSkin_M.cgfx

This skin shader based on Lambert illumination model.

Include effect: Diffuse, Specular, Bump, Backlighting, Simple subsurface scattering.

Technique:

SkinM: Standard Skin illumination model.

TangentNormal: Display surface normal of Tangent space.

ObjectNormal: Display surface normal of Object space.

Parameter:

- bumpHeight:** Controls how high the bumps display on the surface.
- specular:** Controls the intensity of shiny highlights on the surface.
- specExpon:** Controls the size of shiny highlights on the surface.
- sssScale:** Controls the intensity of Subsurface scattering.
- backlightScale:** Controls the intensity of backlighting.
- rollOff:** Controls the intensity of falloff for Subsurface scattering.
- diffuseScale:** Controls the intensity and range of Diffuse.
- lightPos:** Input Light position here.
- lightColor:** Controls Light color and intensity.
- ambiColor:** Controls Ambient color of surface, Set to black by default.
- colorMapSampler:** Input Diffuse color map here.
- normalMapSampler:** Input Normal map here.
- specMapSampler:** Input Specular map here.
- sssMapSampler:** Input Subsurface scattering map here.
- backlightMapSampler:** Input Backlighting map here.

For Bump Map Material

The following scripts is for bump effect by Bump map(High map).

These shading scripts same as "For Normal map material series" with effect, besides bump.

Add one parameter "BumpMapSampleSize" for setting Bump map sample resolution, the more value the more detail, but more expensive. Recommend use default size 512*512(equal to resolution of bump map is better).

[cgBlinn_B_M.cgfx](#)

Refer cgBlinn_M.cgfx

[cgAdBlinn_B_M.cgfx](#)

Refer cgAdBlinn_M.cgfx

[cgMetal_B_M.cgfx](#)

Refer cgMetal_M.cgfx

[cgGlass_B_M.cgfx](#)

Refer cgGlass_M.cgfx

[cgSkin_B_M.cgfx](#)

Refer cgSkin_M.cgfx

Sample Scene and Information



[sample4CgFX.ma](#)

The sample scene explain basic use and effect for CgFXSL Package.

Important information about the CG language including the limitations of the CgFX plugin are available on the [NVIDIA](#) website.

The more detail and update for CgFXSL for Maya, please browse [ASDN](#) Website.

Acknowledgments

Sincere thanks goes to **Yongqing Ye** for his translation.

by Li Liu

If you have any question please browse [ASDN](#) or [Email to me](#).