



***n*VIDIA®**

Sébastien Dominé, NVIDIA

CgFX

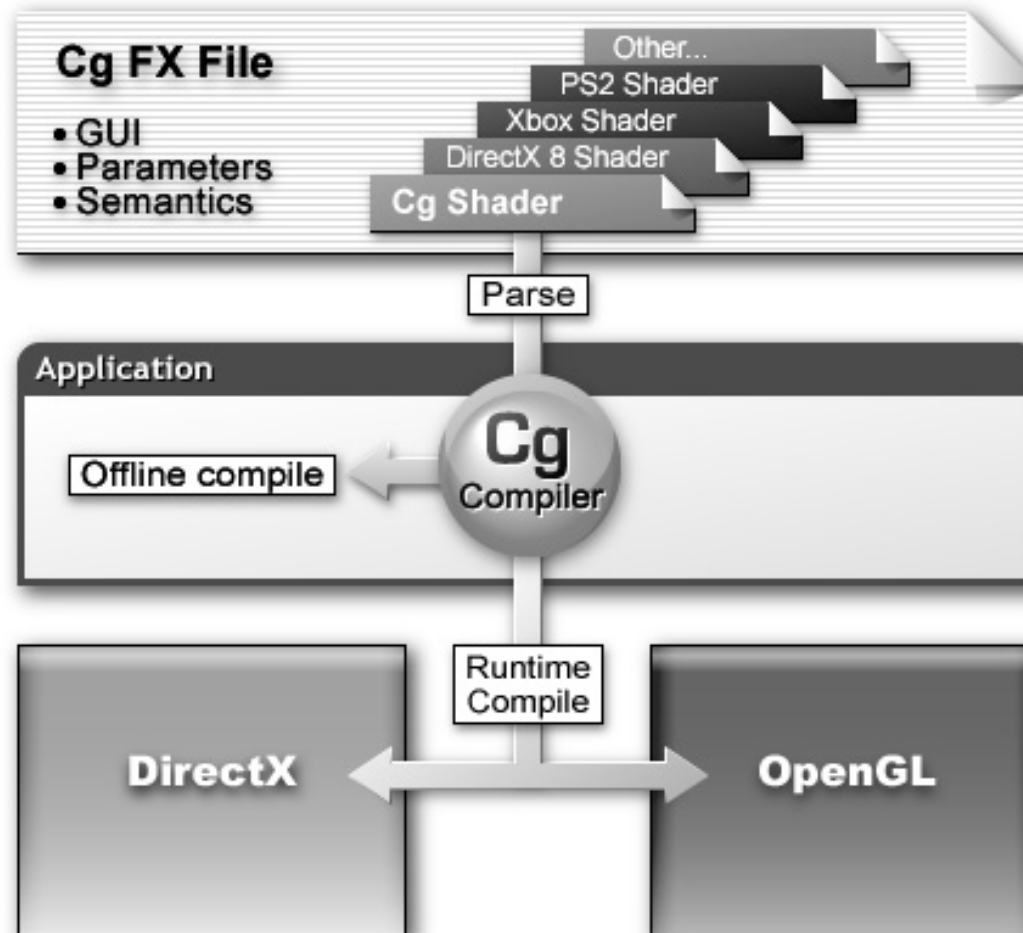
Overview

- **What is CgFX?**
- **CgFX runtime**
- **Production pipeline with CgFX**
- **CgFX Tools set**
- **Demo**

What is CgFX?

- **Supports Microsoft .fx files**
- **Cg plus:**
 - **Multi-pass**
 - **Hardware fallbacks (techniques)**
 - **Complete Hardware states**
 - **Tweakables**
- **MS .fx plus:**
 - **DirectX8 and OpenGL**
 - **MAC and Linux**

CgFX overview



.fx file

```
float4x4 worldMatrix : World;           // World or model matrix
float4x4 mvpMatrix : WorldViewProjection; // Model * View * Projection
float4x4 worldViewMatrix : WorldView;    // World * View

texture diffuseTexture : DiffuseMap      // Diffuse Map semantic
<
    string File = "default_color.dds";   // Default texture file annotation
>;

texture normalMap : NormalMap            // Normal Map semantic
<
    string File = "default_bump_normal.dds"; // Default texture file annotation
>;

float4 ambientColor : Ambient < > = {0.1, 0.1, 0.1, 1.0};

float bumpHeight
<                                     // GUI annotations
    string gui = "slider";
    float min = 0;
    float max = 1;
    float step = 0.1;
> = 0.5;
```

.fx file

```
void DiffuseBumpVS(   float4 Position : POSITION,      // Position in object space
                    float2 TexCoord : TEXCOORD0,    // Texture coordinates
                    ""
                    out float4 TexCoord0 : TEXCOORD0, // Texture coordinates
                    out float4 Position : POSITION) // Position in projection space
{
    TexCoord0.xy = IN.TexCoord.xy;                // Pass texture coordinates for the diffuse map
    ""
    Position = mul(WorldViewProj, IN.Position);    // Compute position in projection space
}

void DiffuseBumpPS(   float4 Position : POSITION,      // Position in projection space
                    ""
                    out float4 col : COLOR)
{
    float4 color = tex2D(DiffuseMap, UV); // Look up the diffuse map
    ""
    col = color * light;                  // Modulate the diffuse color by the light
}
```

.fx file

```
sampler2D diffuseSampler = sampler_state {
    Texture = <diffuseTexture>;
    MinFilter = Linear;
    MagFilter = Linear;
    MipFilter = Linear;
};

sampler2D normalSampler = sampler_state {
    Texture = <normalMap>;
    MinFilter = Linear;
    MagFilter = Linear;
    MipFilter = Linear;
};

technique CgTechnique // Both the vertex and the fragment shaders are in Cg
{
    pass p0
    {
        ZEnable = true;
        ZWriteEnable = true;
        CullMode = None;

        VertexShader = compile vs_1_1 DiffuseBumpVS(mvpMatrix, WorldIMatrix, lightPos);
        PixelShader = compile ps_1_1 DiffuseBumpPS(diffuseSampler, normalSampler, bumpHeight);
    }
    pass p1
    {
        ...
    }
}
```

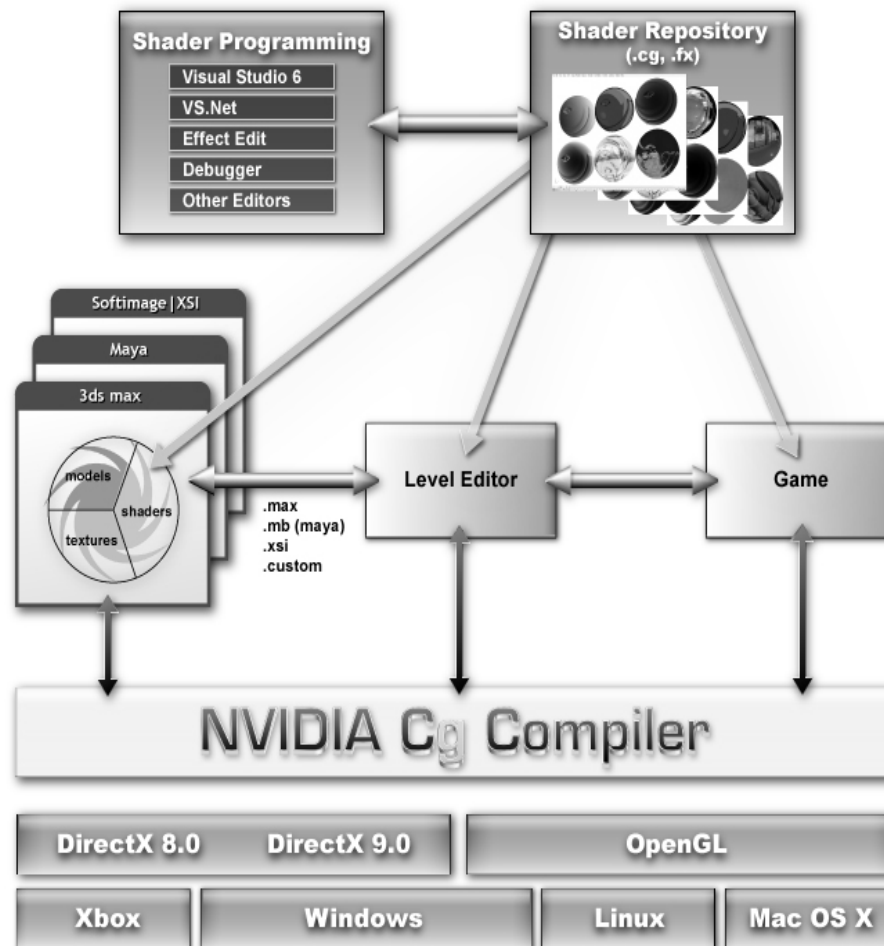
CgFX Runtime

- **Currently at Beta 3**
 - Final release will be fully compatible with DX9 D3DXEffects
- **Simple API enables creation of effects from .fx files**
 - Compiled into efficient state-blocks
- **Can then enumerate techniques, parameters, semantics, etc.**

- **Typical render loop:**

```
pEffect->Begin(&numPasses,0);  
for (PassNum = 0; PassNum < numPasses; ++PassNum)  
{  
    pEffect->Pass(PassNum);  
    pDevice->DrawPrimitive(NVPRIMITIVE_TRIANGLELIST, StartVertex, mesh.num_faces);  
}  
pEffect->End();
```


Production Pipeline with CgFX



CgFX Tools Set

- **Integrated authoring in DCC apps:**
 - 3ds MAX 5.1
 - MAYA 4.5
 - XSI (CgFX Coming soon...)
- **NVB Exporter**
- **CgFX Viewer**
 - OpenGL ARB, DirectX8, DirectX9

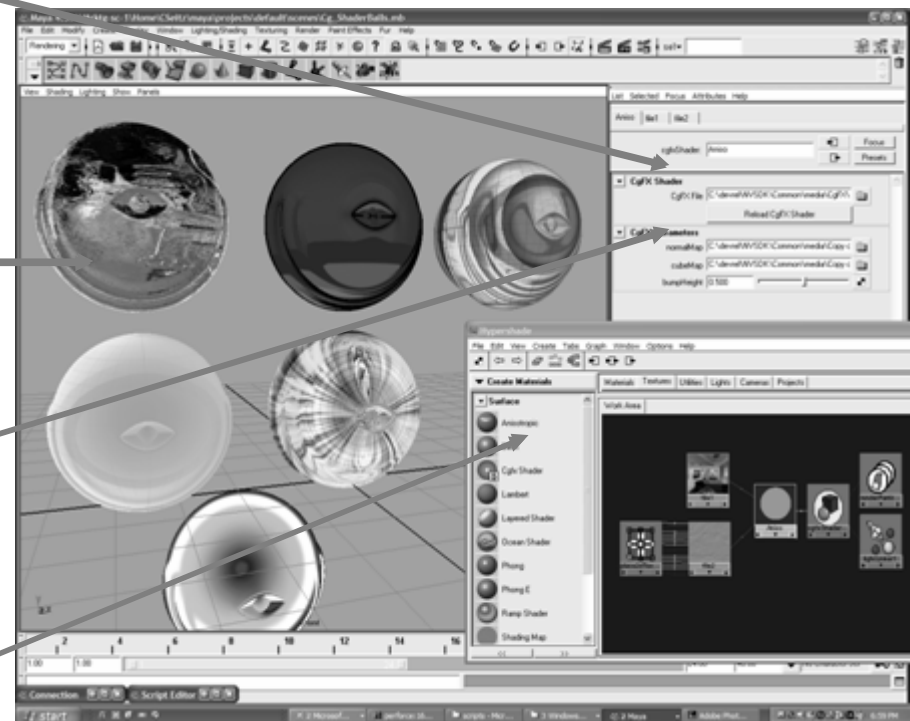
3ds max 5.1

- stdmaterial <-> CgFX
- Height maps to normal maps
- Ability to select MAX scene lights and connect them to .fx parameters
- On the fly editing of shaders and auto-update of .fx GUI
- MAXSCRIPT support
- Source Code



Maya 4.5

- **Dynamic Shader specific GUI**
 - Supports .fx shader format
 - Support native Maya attributes to render .fx tweakables
- **Samples include:**
 - Bumpy Shiny, Toon, Anisotropic Metal, Ghostly, Refraction Dispersion, Rainbow
- **Integrated with Maya's lights**
- **Intuitive artist controls**
 - Slider control over key real-time parameters (e.g., bump depth)
- **CgFX integrated with Maya's hypershade – node based shader editor**
- **Source Code**



NVB Exporter for 3ds max

- **Based on Pierre Terdiman's 3ds max exporter**
 - <http://www.codercorner.com/Flexporter.htm>
- **Exports Scene data**
 - Mesh, materials, lights, camera, skinning, etc...
- **Exports CgFX materials**
 - ICgFXDataBridge interface
- **Source code**

CgFX Viewer

- Scene graph GUI
- .fx parameters edition
- Error reporting for easy .fx file problem identification
- Runs OpenGL, DirectX8, DirectX9
 - Switch between devices at any point



Demos

Questions?

- **Sébastien Dominé**
sdomine@nvidia.com
cgsupport@nvidia.com