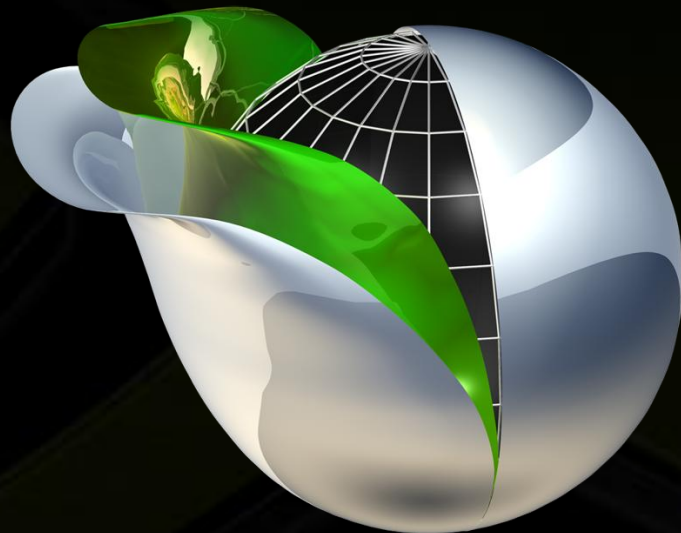




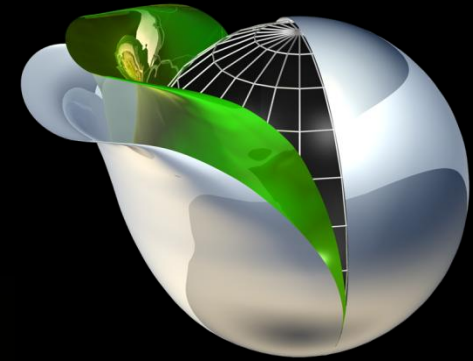
The NVIDIA Shader Debugger



www.shaderdebugger.com

NVIDIA Shader Debugger

Visualize your shaders, step by step



Broad Language Support

• HLSL10/9

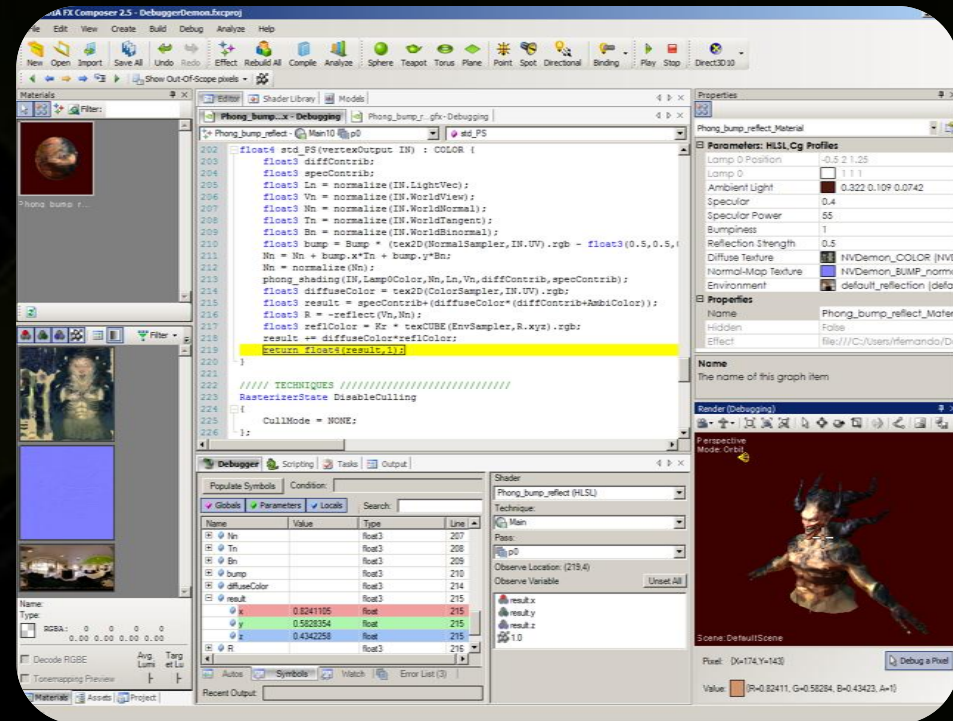
• CgFX

• COLLADA FX Cg

Step through shader source code

Visualize variables across your geometry

Plug-in for FX Composer 2.5



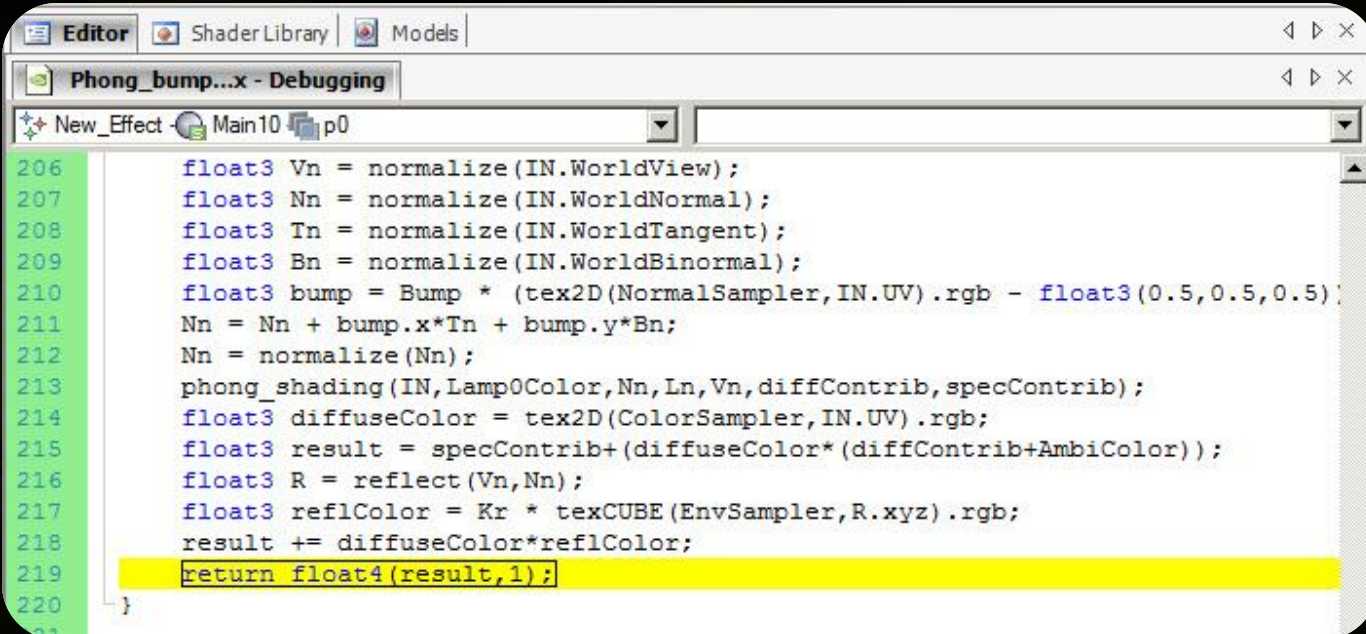
Run Control

Full control of shader execution

Next/Previous Statement

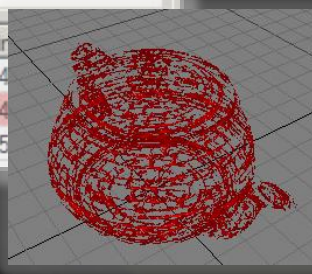
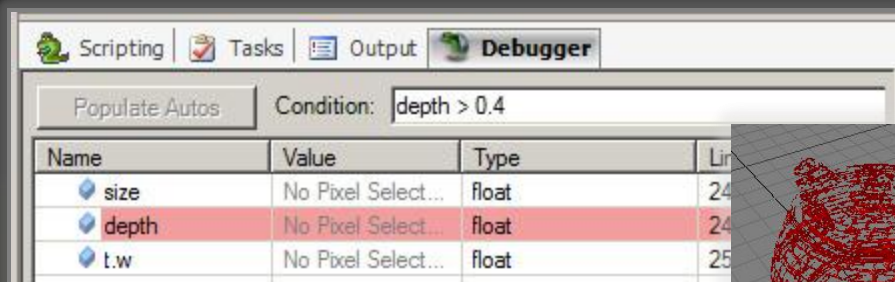
Run to Cursor

Run to Bookmark

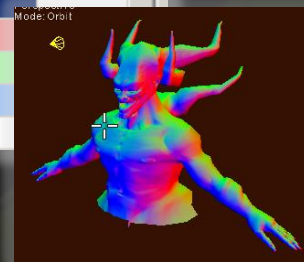
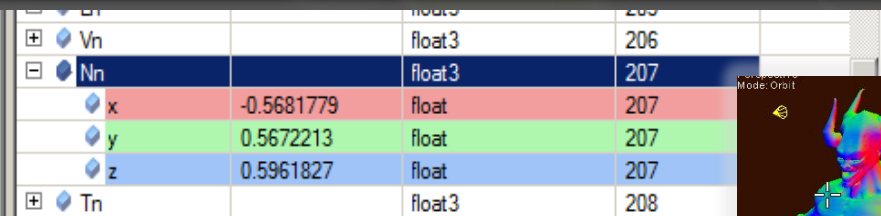


```
Editor | Shader Library | Models
Phong_bump...x - Debugging
New_Effect - Main10 p0
206 float3 Vn = normalize(IN.WorldView);
207 float3 Nn = normalize(IN.WorldNormal);
208 float3 Tn = normalize(IN.WorldTangent);
209 float3 Bn = normalize(IN.WorldBinormal);
210 float3 bump = Bump * (tex2D(NormalSampler, IN.UV).rgb - float3(0.5, 0.5, 0.5));
211 Nn = Nn + bump.x*Tn + bump.y*Bn;
212 Nn = normalize(Nn);
213 phong_shading(IN, Lamp0Color, Nn, Ln, Vn, diffContrib, specContrib);
214 float3 diffuseColor = tex2D(ColorSampler, IN.UV).rgb;
215 float3 result = specContrib + (diffuseColor * (diffContrib + AmbiColor));
216 float3 R = reflect(Vn, Nn);
217 float3 reflColor = Kr * texCUBE(EnvSampler, R.xyz).rgb;
218 result += diffuseColor * reflColor;
219 return float4(result, 1);
220 }
```

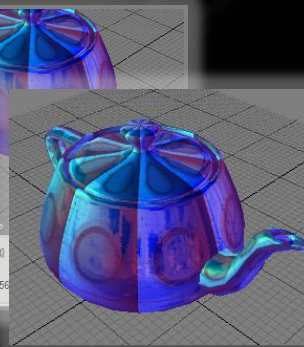
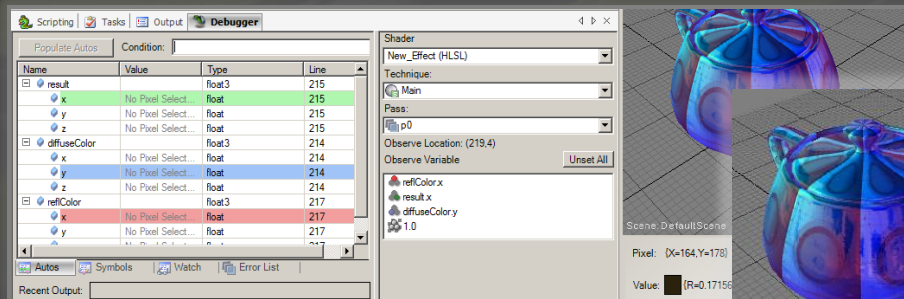
More Shader Debugger Features



Conditionally Kill Fragments



Visualize Any Variable

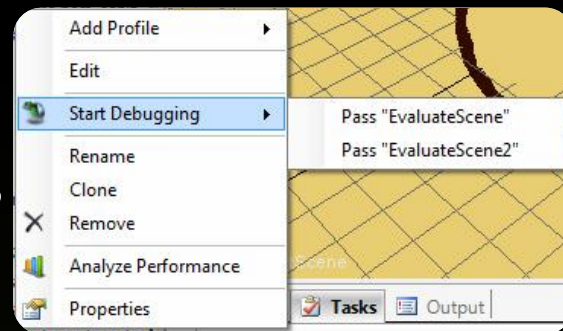


Create Custom Output Mappings

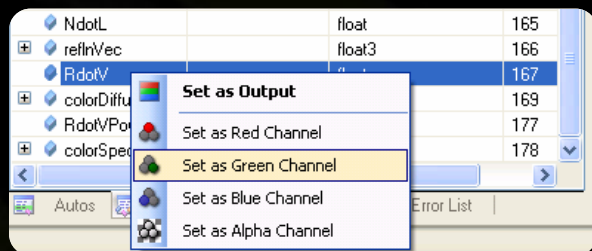
More Shader Debugger Features



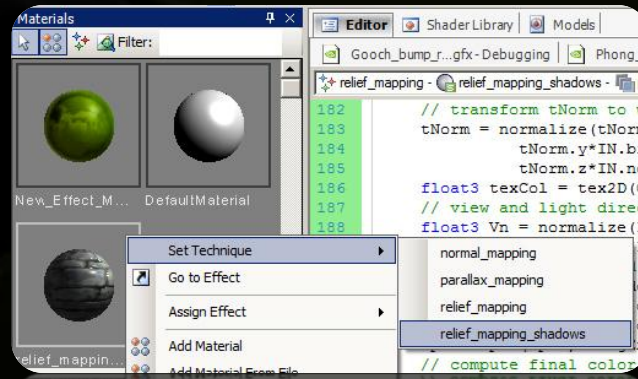
Debug multi-pass/full scene effects.



Map any variable to any channel.



Pick any technique in your shader to debug.



The NVIDIA Software Improvement Program (SIP)



- **Goal: Better meet developer needs**
 - Simplify common usage patterns
 - Expand on popular features
 - Highlight powerful but underutilized features
 - Remove unwanted features
- **Instant Feedback allows user to press F4 and directly send a comment to NVIDIA**
- **Developer opts in to the SIP to participate – we encourage you to do so**
- **Only information about product features used is gathered, as well as GPU and driver version.**
- **No personally identifiable data, shaders, textures, scripts, geometry, or information about other applications is ever collected.**

Trying Out The NVIDIA Shader Debugger



- **Read the Quick Tutorial**
 - This will guide you through the basics quickly
- **Read the User Guide**
 - This explains all product features in detail
 - Starts from the basics and moves to advanced topics
- **Let us know what you think**
 - Don't be shy with Instant Feedback (F4)!
 - Send feedback to ShaderDebugger-Support@nvidia.com