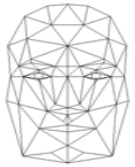
Debugging

There is no debugger! You must use other tricks!

- **Compiler error messages**
- **Signals using vertex shader**
- **Signals using fragment shader**
- **Use simple geometry - easy to understand**

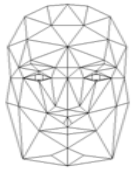


InfoLog

glGetProgramInfoLog/glGetShaderInfoLog (glGetInfoLogARB)

Retrieves information about compilation and linking results

May include error messages, warnings... The exact contents varies depending on GPU brand.



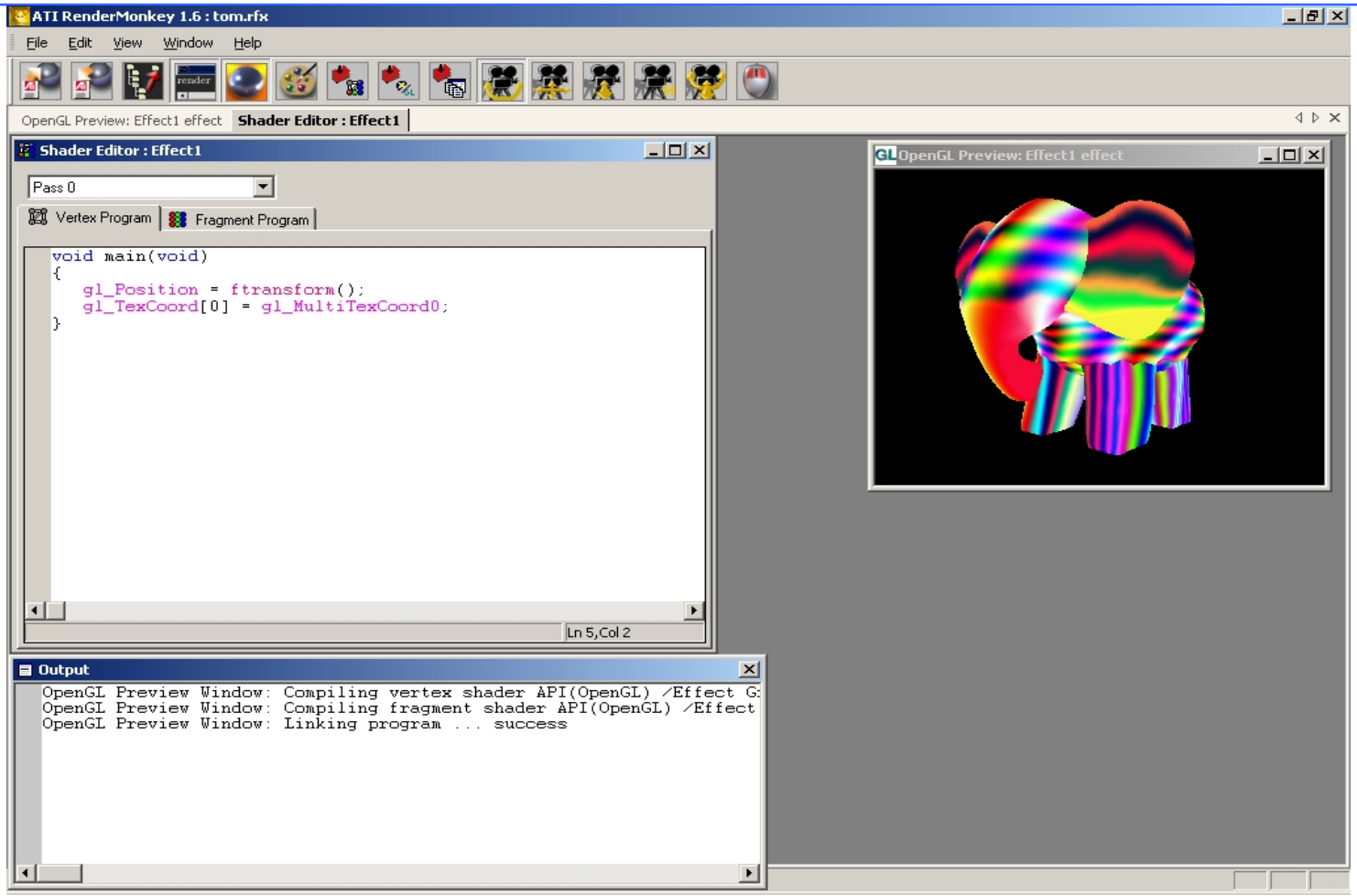
Development tools

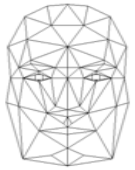
Shader development directly in large application unreasonable

Simple development shells are used to:

- **Edit source code for vertex and fragment shader**
- **Recompile when desired**
- **Test the shader on a model**
- **Display compilation results**

I.e. Rendermonkey, OpenGL Shader Builder, and our lab shell for lab 3


ATI RenderMonkey 1.6 : tom.rfx
File Edit View Window Help
OpenGL Preview: Effect1 effect
Shader Editor : Effect1
Shader Editor : Effect1
Pass 0
Vertex Program
Fragment Program
void main(void)
{
gl_Position = ftransform();
gl_TexCoord[0] = gl_MultiTexCoord0;
}
Ln 5,Col 2
OpenGL Preview: Effect1 effect
Output
OpenGL Preview Window: Compiling vertex shader API(OpenGL) /Effect G:
OpenGL Preview Window: Compiling fragment shader API(OpenGL) /Effect
OpenGL Preview Window: Linking program ... success



The shader lab shell

Created by Lars Abrahamsson, more or less rewrittenthis year by Mikael Kalms.

- **Source code edited in separate editor, as two files put in a certain place.**
- **Compilation results in the shell window**
- **Two pre-installed textures**
- **Can switch between shader and fixed pipeline**

Information Coding / Computer Graphics, ISY, LiTH