

WebGLU Development Library for WebGL

Benjamin P. DeLillo*
Rochester Institute of Technology

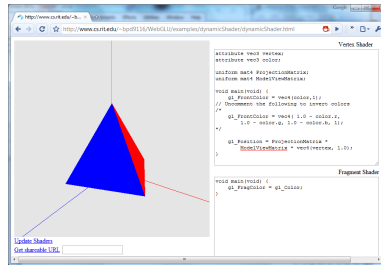


Figure 1: Dynamic Shader Editing

1 Introduction

For more than a decade, rich 3D content on the web has only been available via external, often proprietary, browser plugins. However, a new standard has emerged to change this. WebGL, currently under development by the Khronos Group, is a standard specification for javascript bindings to OpenGL [Khronos 2009]. In September 2009 WebGL support made its way to development builds of Firefox 3.7. Since this time, WebGL has gained greater traction and visibility within developer communities. Although impressive demonstrations of WebGL are available [Vukicevic 2009], we believed that the creation of a development library would help kick-start interest in the creation of new applications.

In this paper, we present WebGLU, a development library which mitigates the sometimes esoteric complexity of real-time 3D by providing a clean but powerful API. WebGLU's design goals of separating rendering concerns from the overall design of an application, simplifying the process of implementation, and making the resulting application code more concise allows developers to spend more time focused on producing compelling content.

WebGLU is open source under the MIT license, the library itself, some examples and more information is available at <http://github.com/OneGeek/WebGLU>. Live examples are maintained at <http://www.cs.rit.edu/~bpd9116/WebGLU/examples>.

2 Features

The primary WebGLU `Object` class maintains arrays of shader attribute data and the associated WebGL buffers, it also keeps track of any textures. The `Object` will dynamically bind those buffers and textures before rendering. `Objects` can be associated hierarchically and assigned procedural or keyframed animations, even mixing animation types within a single hierarchy. In progress developments include both physical simulation and behaviorally generated animation.

WebGLU's `Shader` and `ShaderProgram` classes parse out shader attributes and uniforms before compiling and automatically associates common uniforms, such as the `Model-View` and `Projection` matrices, with the per-frame actions that must be taken for each. Custom actions are also supported. `Compile` and `Link` status are also tracked, with compilation and linking done as needed.

*e-mail: Benjamin.DeLillo@mail.rit.edu

3 The WebGLU Difference

When working with traditional WebGL just getting data into a shader is verbose and tedious. As the code below illustrates, moving data to a shader can be tedious. This varies for different shaders, shader programs, and sets of data. This results in a brittle system that requires modification to incorporate even small changes. WebGLU eliminates these drawbacks and allows for faster and more efficient WebGL development.

```
gl.bindBuffer(gl.ARRAY_BUFFER, myVertexBuffer);
gl.bufferData(gl.ARRAY_BUFFER, myVertexDataArray,
             gl.STATIC_DRAW);
gl.vertexAttribPointer(gl.getAttribLocation(
                      gl.getAttribLocation(myShaderProgram, "vertex"),
                      3, gl.FLOAT, false, 0, 0);
gl.enableVertexAttribArray(gl.getAttribLocation(
gl.getAttribLocation(myShaderProgram, "vertex"));
```

WebGLU makes things much more easy and unified, keeping track of all the buffers, attributes, uniforms, and ensures everything that needs to be called at render time is called.

```
anObject = new $W.Object($W.GL.TRIANGLES);
anObject.fillArray("vertex", myVertexDataArray);
anObject.vertexCount = myVertexDataArray.length;
```

4 Future Work

Future features include support for complex multi-pass rendering and object picking. It will be possible to harness the connected nature of the internet to build traditional 3D content tools like model, animation, and shader editors which work collaboratively.

Acknowledgements

To Reynold Bailey and Joe Geigel for their support and guidance.

References

- KHRONOS, 2009. Khronos details webgl initiative to bring hardware-accelerated 3d graphics to the internet. <http://www.khronos.org/news/press/releases/khronos-webgl-initiative-hardware-accelerated-3d-graphics-internet/August>.
- VUKICEVIC, V., 2009. Webgl samples/demos and other bits. <http://blog.vladil.com/2009/09/21/webgl-samples/>, September.