Introduction to Computer Graphics with WebGL

Ed Angel
Professor Emeritus of Computer Science
Founding Director, Arts, Research, Technology and Science Laboratory
University of New Mexico
Programming with WebGL
Part 1: Background

Ed Angel
Professor Emeritus of Computer Science
University of New Mexico
OpenGL Architecture
Software Organization

The University of New Mexico

Angel and Shreiner: Interactive Computer Graphics 7E © Addison-Wesley 2015
A OpenGL Simple Program

Generate a square on a solid background
#include <GL/glut.h>
void mydisplay(){
    glClear(GL_COLOR_BUFFER_BIT);
    glBegin(GL_QUAD);
    glVertex2f(-0.5, -0.5);
    glVertex2f(-0.5, 0.5);
    glVertex2f(0.5, 0.5);
    glVertex2f(0.5, -0.5);
    glEnd()
}
int main(int argc, char** argv){
    glutCreateWindow("simple");
    glutDisplayFunc(mydisplay);
    glutMainLoop();
}
What happened?

• Most OpenGL functions deprecated
  - immediate vs retained mode
  - make use of GPU

• Makes heavy use of state variable default values that no longer exist
  - Viewing
  - Colors
  - Window parameters

• However, processing loop is the same
Execution in Browser

Browser

Web Server

URL

Web Page

HTML
JS files

JS Engine

CPU/GPU

Framebuffer

Canvas
Event Loop

• Remember that the sample program specifies a render function which is an event listener or callback function
  - Every program should have a render callback
  - For a static application we need only execute the render function once
  - In a dynamic application, the render function can call itself recursively but each redrawing of the display must be triggered by an event
Lack of Object Orientation

• All versions of OpenGL are not object oriented so that there are multiple functions for a given logical function

• Example: sending values to shaders
  - `gl.uniform3f`
  - `gl.uniform2i`
  - `gl.uniform3dv`

• Underlying storage mode is the same
WebGL function format

```
void function (uint32_t x, float y, float z)
```

- `x`, `y`, `z` are floats
- `function` is a function name
- `uint32_t` is a dimension

```
float* ptr = &x;
gl.uniform3fv(ptr);  // is an vector
```
WebGL constants

• Most constants are defined in the canvas object
  - In desktop OpenGL, they were in #include files such as gl.h

• Examples
  - desktop OpenGL
    • glEnable(GL_DEPTH_TEST);
  - WebGL
    • gl.enable(gl.DEPTH_TEST)
  - gl.clear(gl.COLOR_BUFFER_BIT)
WebGL and GLSL

- WebGL requires shaders and is based less on a state machine model than a data flow model.
- Most state variables, attributes and related pre 3.1 OpenGL functions have been deprecated.
- Action happens in shaders.
- Job of application is to get data to GPU.
GLSL

• OpenGL Shading Language
• C-like with
  - Matrix and vector types (2, 3, 4 dimensional)
  - Overloaded operators
  - C++ like constructors
• Similar to Nvidia’s Cg and Microsoft HLSL
• Code sent to shaders as source code
• WebGL functions compile, link and get information to shaders