

# SGI – WebGL

## Uniforms and matrices

Nuria Pelechano Gómez  
npelechano@lsi.upc.edu

- ◉ Learning WebGL:

- > [http://learningwebgl.com/blog/?page\\_id=1217](http://learningwebgl.com/blog/?page_id=1217)

- ◉ Lesson 1: Lesson 1: A Triangle and a Square gives you an overview of how WebGL works, with enough code to simply draw a static triangle and a square on the screen.

# Include matrices library

```
<script type="text/javascript" src="glMatrix-0.9.5.min.js"></script>
```

# FS with uniforms

- Vertex matrix transformations

```
<script id="shader-vs" type="x-shader/x-vertex">
attribute vec3 aVertexPosition;

uniform mat4 uMVMatrix;
uniform mat4 uPMatrix;

void main(void) {
    gl_Position = uPMatrix * uMVMatrix * vec4(aVertexPosition, 1.0);
}

</script>
```

# Get uniform location

```
function initShaders() {  
    ...  
    shaderProgram.pMatrixUniform =  
gl.getUniformLocation(shaderProgram, "uPMatrix");  
    shaderProgram.mvMatrixUniform =  
gl.getUniformLocation(shaderProgram, "uMVMatrix");  
  
    ...  
}
```

# Create your vbles in js

```
var mvMatrix = mat4.create();  
var pMatrix = mat4.create();  
  
function setMatrixUniforms() {  
    gl.uniformMatrix4fv(shaderProgram.pMatrixUniform, false, pMatrix);  
    gl.uniformMatrix4fv(shaderProgram.mvMatrixUniform, false,  
mvMatrix);  
}
```

# Assign uniform values and pass to GPU

```
function drawScene() {  
    gl.viewport(0, 0, gl.viewportWidth, gl.viewportHeight);  
    gl.clear(gl.COLOR_BUFFER_BIT | gl.DEPTH_BUFFER_BIT);  
  
    mat4.perspective(45, gl.viewportWidth / gl.viewportHeight, 0.1, 100.0,  
pMatrix);  
  
    mat4.identity(mvMatrix);  
    mat4.translate(mvMatrix, [-1.5, 0.0, -7.0]);  
  
    gl.bindBuffer(gl.ARRAY_BUFFER, triangleVertexPositionBuffer);  
    gl.vertexAttribPointer(shaderProgram.vertexPositionAttribute,  
triangleVertexPositionBuffer.itemSize, gl.FLOAT, false, 0, 0);  
  
    setMatrixUniforms();  
    gl.drawArrays(gl.TRIANGLES, 0, triangleVertexPositionBuffer.numItems);  
    ...  
}
```