




08-10 JULY 2014
Develop
in Brighton

Join the conversation

 @developconf #developconf

 facebook.com/DevelopConference

 Network: developconf@event
Password: mayalt

Download the Develop Conference 2014 App

- download at developconference.com
- check into this session
- don't forget to give us your feedback

FREE WiFi access for all delegates!

Go to BT Openzone, go to "Log on with your conference details":

Network: developconf@event Password: mayalt

Sponsored by



The logo for WebGL, featuring the text "WebGL" in a bold, red, sans-serif font. The "W" is stylized with a thick, curved stroke that loops around the top and bottom of the letters "e" and "b". A small "TM" trademark symbol is located to the right of the "L".

WebGL™

WTF or FTW

An introduction



08-10 JULY 2014

Develop
in Brighton

Join the conversation

 @developconf #developconf

 facebook.com/DevelopConference

Carl Bateman



- By day --
 - Software Engineer (desktop)
 - C#, C++, VB, MySQL, .NET, Linq, blah, blah, blah
- By night --
 - Software Engineer (misc)
 - OpenGL, Unity, JavaScript, PHP, CSS, HTML and, of course, WebGL
- Love 3d and game jams

Carl Bateman



WebGL Workshop

(free, monthly)

SkillsMatter, London



08-10 JULY 2014
Develop
in Brighton

Join the conversation

 @developconf #developconf

 facebook.com/DevelopConference

WebGL

- Powerful
- Growing
- Widely Supported
- Widely Used
 - PS4 gui



08-10 JULY 2014

Develop
in Brighton

Join the conversation

 @developconf #developconf

 facebook.com/DevelopConference

Agenda

- Background and history
- Browser and hardware support
- Basic coding and setup
- Security issues
- APIs
- Dev tools
- Debugging
- Support
- Language features

Demo 1

- Epic Citadel
- Unreal Engine ported in 3 days from C++ using Emscripten

<https://www.youtube.com/watch?v=MyAcKgr6mZU>

Background and history

OpenGL ES2.0 for the web

Developer

Vladimir Vukićević

Mozilla

March 2011

Browser support

IE	Firefox	Chrome	Safari	Opera	iOS Safari	Opera Mini
11.0	30.0 Partial	35.0 Supported	7.0 Partial	22.0 Supported	8.0	5.0-7.0 Unsupported

Android Browser	Blackberry Browser	Opera Mobile	Chrome Android	Firefox Android	IE Mobile
4.4 Unsupported	10.0 Supported	22.0 Supported	35.0 Partial	30.0 Partial	10.0 Unsupported

Browser support

get.Webgl.org

caniuse.com/webgl

browserleaks.com/webgl

webglreport.com

doesmybrowsersupportwebgl.com

WebGL

OpenGL ES 2.0 for the web

GLSL

JavaScript

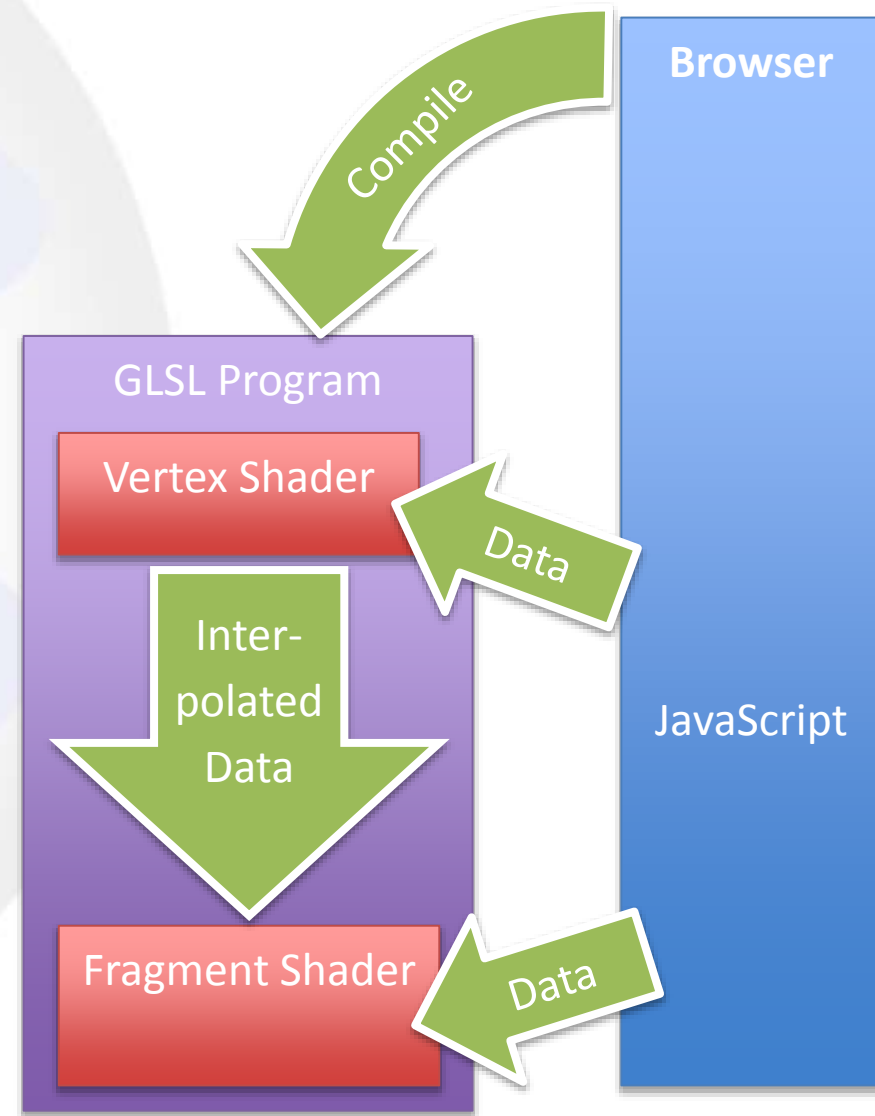
HTML

CSS

WebGL

Built-in API

GPU access via
GLSL program



08-10 JULY 2014
Develop
in Brighton

Join the conversation

 @developconf #developconf

 facebook.com/DevelopConference

Demo 2

<http://www.babylonjs.com/index.html?TRAIN>



08-10 JULY 2014

Develop
in Brighton

Join the conversation



@developconf #developconf



facebook.com/DevelopConference

Basic coding and setup

1. Write a vertex and fragment shader
2. Create a context
3. Initialise the shaders
4. Initialise the engine
5. Set up the viewport
6. Set up data
7. Get variable location in shaders
8. Send data to shaders
9. Render

WebGL

Raw WebGL can be Verbose



08-10 JULY 2014

Develop
in Brighton

Join the conversation

 @developconf #developconf

 facebook.com/DevelopConference

Write a vertex shader

```
<script id="vertex" type="x-shader">
  attribute vec3 aVertexPosition;
  attribute vec4 aVertexColor;

  uniform mat4 uMVMatrix;
  uniform mat4 uPMatrix;

  varying vec4 vColor;

  void main(void) {
    gl_Position = uPMatrix * uMVMatrix *
vec4(aVertexPosition, 1.0);
    vColor = aVertexColor;
  }
</script>
```

Write a fragment shader

```
<script id="fragment" type="x-shader">
  precision highp float;
  varying vec4 vColor;

  void main() {
    gl_FragColor = vColor;
  }
</script>
```

Initialise WebGL

```
var canvas = document.getElementById("mycanvas");
var names = ["webgl", "experimental-webgl",
             "webkit-3d", "moz-webgl"];
for (var i = 0; i < names.length; ++i) {
  try {
    gl = canvas.getContext(names[i]);
  }
  catch (e) { }
  if (gl) break;
}

gl.viewportWidth = canvas.width;
gl.viewportHeight = canvas.height;
```

Initialise Shader Program

```
var v =  
document.getElementById("vertex").firstChild.nodeValue;  
var f =  
document.getElementById("fragment").firstChild.nodeValue;  
  
var vs = gl.createShader(gl.VERTEX_SHADER);  
gl.shaderSource(vs, v);  
gl.compileShader(vs);  
  
var fs = gl.createShader(gl.FRAGMENT_SHADER);  
gl.shaderSource(fs, f);  
gl.compileShader(fs);  
  
shaderProgram = gl.createProgram();  
gl.attachShader(shaderProgram, vs);  
gl.attachShader(shaderProgram, fs);  
gl.linkProgram(shaderProgram);
```

Get Variable Locations

```
shaderProgram.pMatrixUniform =  
gl.getUniformLocation(shaderProgram, "uPMatrix");  
shaderProgram.mvMatrixUniform =  
gl.getUniformLocation(shaderProgram, "uMVMatrix");  
  
shaderProgram.vertexPositionAttribute =  
gl.getAttribLocation(shaderProgram,  
                      "aVertexPosition");  
gl.enableVertexAttribArray(shaderProgram  
                           .vertexPositionAttribute);  
shaderProgram.vertexColorAttribute =  
gl.getAttribLocation(shaderProgram, "aVertexColor");  
gl.enableVertexAttribArray(shaderProgram  
                           .vertexColorAttribute);
```

Define Vertices

```
var vertices = new Float32Array([ ... ]);
cubeVertexPositionBuffer = gl.createBuffer();
gl.bindBuffer(gl.ARRAY_BUFFER,
               cubeVertexPositionBuffer);
gl.bufferData(gl.ARRAY_BUFFER, vertices,
              gl.STATIC_DRAW);
cubeVertexPositionBuffer.itemSize = 3;
cubeVertexPositionBuffer.numItems =
    vertices.length/vertexBuffer.itemSize;
```

Define Vertex Colours

```
var colors = new Float32Array([ ... ]);
cubeVertexColorBuffer = gl.createBuffer();
gl.bindBuffer(gl.ARRAY_BUFFER,
              cubeVertexColorBuffer);
gl.bufferData(gl.ARRAY_BUFFER, colors,
              gl.STATIC_DRAW);
cubeVertexColorBuffer.itemSize = 4;
cubeVertexColorBuffer.numItems = 24;
```

Define Indices

```
var indices = new Uint16Array([ ... ]);
cubeVertexIndexBuffer = gl.createBuffer();
gl.bindBuffer(gl.ELEMENT_ARRAY_BUFFER,
              cubeVertexIndexBuffer);
gl.bufferData(gl.ELEMENT_ARRAY_BUFFER, indices,
              gl.STATIC_DRAW);
cubeVertexIndexBuffer.itemSize = 1;
cubeVertexIndexBuffer.numItems = 36;
```


Render - 1

```
mat4.perspective(45, gl.viewportWidth /  
                gl.viewportHeight, 0.1, 100.0, pMatrix);  
mat4.identity(mvMatrix);  
mat4.translate(mvMatrix, [0.0, 0.0, -4.0]);  
mat4.multiply(mvMatrix, rotationMatrix);  
  
gl.uniformMatrix4fv(shaderProgram.pMatrixUniform,  
                    false, pMatrix);  
  
gl.uniformMatrix4fv(shaderProgram.mvMatrixUniform,  
                    false, mvMatrix);
```

Render - 2

```
gl.bindBuffer(gl.ARRAY_BUFFER,  
              cubeVertexPositionBuffer);  
gl.vertexAttribPointer(shaderProgram.vertexPositionA  
ttribute, cubeVertexPositionBuffer.itemSize,  
gl.FLOAT, false, 0, 0);  
  
gl.bindBuffer(gl.ARRAY_BUFFER,  
              cubeVertexColorBuffer);  
gl.vertexAttribPointer(shaderProgram.vertexColorAttr  
ibute, cubeVertexColorBuffer.itemSize, gl.FLOAT,  
false, 0, 0);  
gl.bindBuffer(gl.ELEMENT_ARRAY_BUFFER,  
              cubeVertexIndexBuffer);
```

Render - 3

```
gl.clearColor(0, 0.5, 0, 1);  
gl.clear(gl.COLOR_BUFFER_BIT |  
         gl.DEPTH_BUFFER_BIT);  
gl.enable(gl.DEPTH_TEST);  
  
gl.drawElements(gl.TRIANGLES,  
                cubeVertexBuffer.numItems,  
                gl.UNSIGNED_SHORT, 0);
```

Animation Loop

```
function tick() {  
  requestAnimationFrame(tick);  
  animate();  
  draw();  
}
```



08-10 JULY 2014

Develop
in Brighton

Join the conversation

 @developconf #developconf

 facebook.com/DevelopConference

Demo 3

Da cube 1

webglworkshop.com/develop2014



08-10 JULY 2014

Develop
in Brighton

Join the conversation

 @developconf #developconf

 facebook.com/DevelopConference

APIs

- Quicker
- e.g. Three.js



08-10 JULY 2014
Develop
in Brighton

Join the conversation

 @developconf #developconf

 facebook.com/DevelopConference

Includes and Variables

```
<script src="three.min.js"></script>  
<script src="OrbitControls.js"></script>
```

```
var scene = new THREE.Scene();  
var camera = new THREE.PerspectiveCamera(75, 1, 0.1,  
1000);  
var renderer = new THREE.WebGLRenderer();  
camera.position.z = 3;  
  
renderer.setSize(500, 500);  
renderer.setClearColor(0x007900, 1);  
  
document.body.appendChild(renderer.domElement);
```

Define Geometry

```
var geometry = new THREE.CubeGeometry(1, 1, 1);
var cubeMaterialArray = [];
cubeMaterialArray.push(new THREE.MeshBasicMaterial(
  { color: 0xff3333 }));
cubeMaterialArray.push(new THREE.MeshBasicMaterial(
  { color: 0xff8800 }));
cubeMaterialArray.push(new THREE.MeshBasicMaterial(
  { color: 0xffff33 }));
cubeMaterialArray.push(new THREE.MeshBasicMaterial(
  { color: 0x33ff33 }));
cubeMaterialArray.push(new THREE.MeshBasicMaterial(
  { color: 0x3333ff }));
cubeMaterialArray.push(new THREE.MeshBasicMaterial(
  { color: 0x8833ff }));
var cubeMaterials = new
  THREE.MeshFaceMaterial(cubeMaterialArray);
var cube = new THREE.Mesh(geometry, cubeMaterials);
scene.add(cube);
```


Controls, Animation Loop and Render

```
var controls = new THREE.OrbitControls(camera,  
renderer.domElement);
```

```
var render = function () {  
    requestAnimationFrame(render);  
    renderer.render(scene, camera);  
    controls.update();  
};  
  
render();
```



08-10 JULY 2014
Develop
in Brighton

Join the conversation

 @developconf #developconf

 facebook.com/DevelopConference

Demo 4

Da cube 2

webglworkshop.com/develop2014



08-10 JULY 2014

Develop
in Brighton

Join the conversation

 @developconf #developconf

 facebook.com/DevelopConference

Some Other APIs

- Babylon.js
- PhiloGL
- SceneJS
- GLGE



08-10 JULY 2014

Develop
in Brighton

Join the conversation



@developconf #developconf



facebook.com/DevelopConference

Dev tools

- Any text editor
 - Vim
 - Notepad++
- Visual Studio
- Brackets.io
- C9.io (cloud based)
- CodeLobster
- Eclipse
- Aptana



08-10 JULY 2014
Develop
in Brighton

Join the conversation

 @developconf #developconf

 facebook.com/DevelopConference

Dev tools

- Made with Marmalade
- PlayCanvas (cloud based)

Debugging

- Console
- WebGL Inspector
- Firefox – Shader Editor
- Error handling

Support - learning

- WebGL.org
- khronos.org/webgl/wiki
- [LearningWebGL](#)
- [LearningThreejs](#)

Support – learning

- LearningWebGL.com
- LearningThreejs.com
- developer.mozilla.org/en-US/docs/Web/WebGL



08-10 JULY 2014

Develop
in Brighton

Join the conversation

 @developconf #developconf

 facebook.com/DevelopConference

Support – shader editors

- www.shadertoy.com
- glsl.heroku.com
- www.kickjs.org/example/shader_editor/shader_editor.html
- codedstructure.net/projects/webgl_shader_lab

Point

- Lot of resources



08-10 JULY 2014

Develop
in Brighton

Join the conversation

 @developconf #developconf

 facebook.com/DevelopConference

Security issues

- Cross origin resource sharing
- Cross domain
- Development
 - Server
 - Switches

Language Features

- JavaScript
- GLSL
- Image loading
- Input handling



08-10 JULY 2014

Develop
in Brighton

Join the conversation

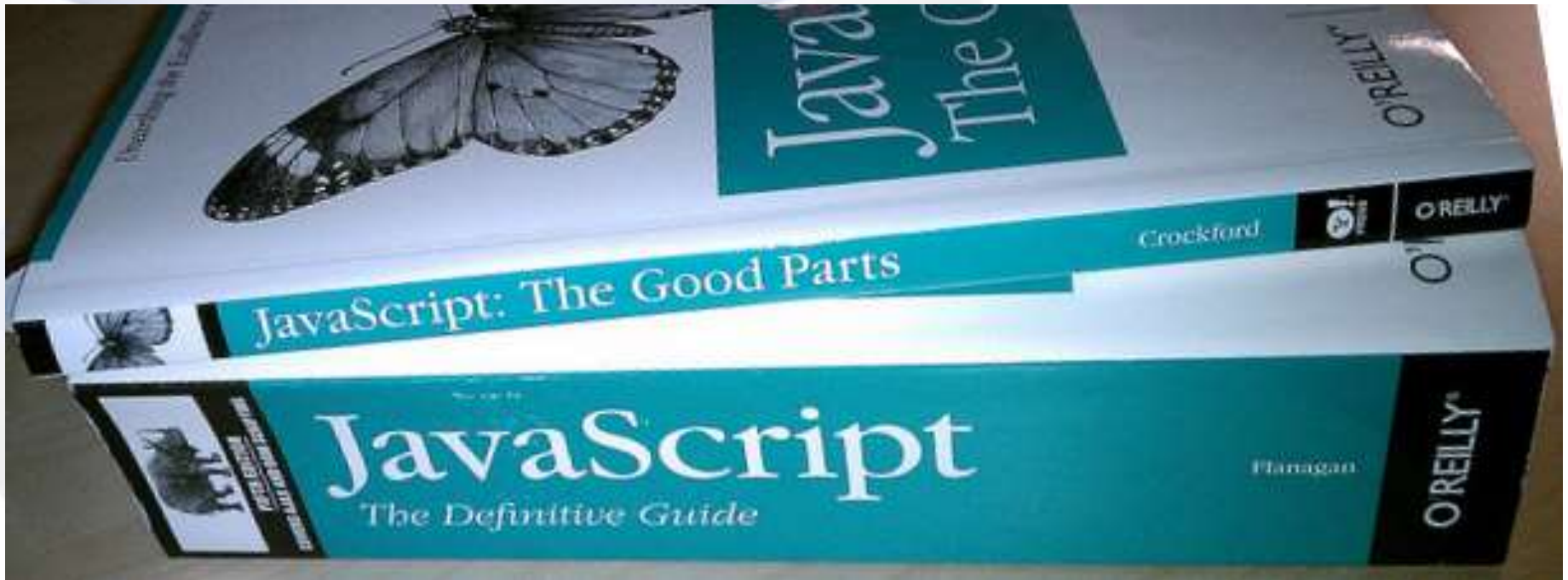


@developconf #developconf



facebook.com/DevelopConference

JavaScript



https://c1.staticflickr.com/5/4066/4704268314_bb0e9d0ff3_z.jpg

JavaScript

- Types (huh?)
- Scope (hoisting)
- Functions – first class objects
- Closures
- Prototypal inheritance
- Objects easily extended

Demos

cubeslam.com

tanks.moka.co



08-10 JULY 2014

Develop
in Brighton

Join the conversation

 @developconf #developconf

 facebook.com/DevelopConference

GLSL

- Vectors
- Matrices
- Attribute
- Uniform
- Varying
- Swizzles (.xyz, .zzz, .rgb, .brg, ~~.xgz~~)



08-10 JULY 2014
Develop
in Brighton

Join the conversation

 @developconf #developconf

 facebook.com/DevelopConference

GLSL

- Dot
- Cross
- Length
- Reflect



08-10 JULY 2014

Develop
in Brighton

Join the conversation

 @developconf #developconf

 facebook.com/DevelopConference



Don't forget to vote



08-10 JULY 2014

Develop
in Brighton

Join the conversation

 @developconf #developconf

 facebook.com/DevelopConference

**Carl.Bateman@
GraphicsCode.com**

**WebGLworkshop.com/
Develop2014**

Questions?