

BREAKING THROUGH 2-DIMENSIONS

INTRODUCTION TO WEBGL

```
1 //Add array iteration method
2 Array.prototype.each = function(f) {
3     var len = this.length;
4     for ( var i = 0; i < len; i++) f(this[i]);
5 };
6
7 //Initializes 3D rendering
8 function initRendering() {
9     "DEPTH_TEST COLOR_MATERIAL LIGHTING LIGHT0 NORMALIZE COLOR_MATERIAL"
10    .split(" ").each(function(elem) {
11        Gl.Enable(Gl[elem]);  OpenGL on V8 baby!
12    });
13 }
14
15 //angle variable
16 var angle = 0;
17
18 //Draws the 3D scene
19 function drawScene() {
20     //Set global color and -STEPHEN HAWKING
21     Gl.Clear(Gl.COLOR_BUFFER
22     GlMatrixMode(Gl.MODELVIEW)
23     GlLoadIdentity();
24     Gl.Translatef(0.0, 0.0,
25     //Set diffuse and position
26     Gl.LightModelfv(Gl.LIGHT0,
27     Gl.Lightfv(Gl.LIGHT0, Gl
28     Gl.Lightfv(Gl.LIGHT0, Gl
29     //Rotate and plot Icosahedron
```





[HTTP://GOO.GL/CA5KLX](http://goo.gl/CA5KLX)

OUTLINE

- What's in a browser (circa HTML5)?
- Defining WebGL
- Brief History
- Why WebGL - Benefits/Disadvantages?
- Applications of WebGL
- The WebGL Pipeline
- Sample Vertex and Fragment Shaders
- How to Get Started
- Debugging
- WebGL Resources

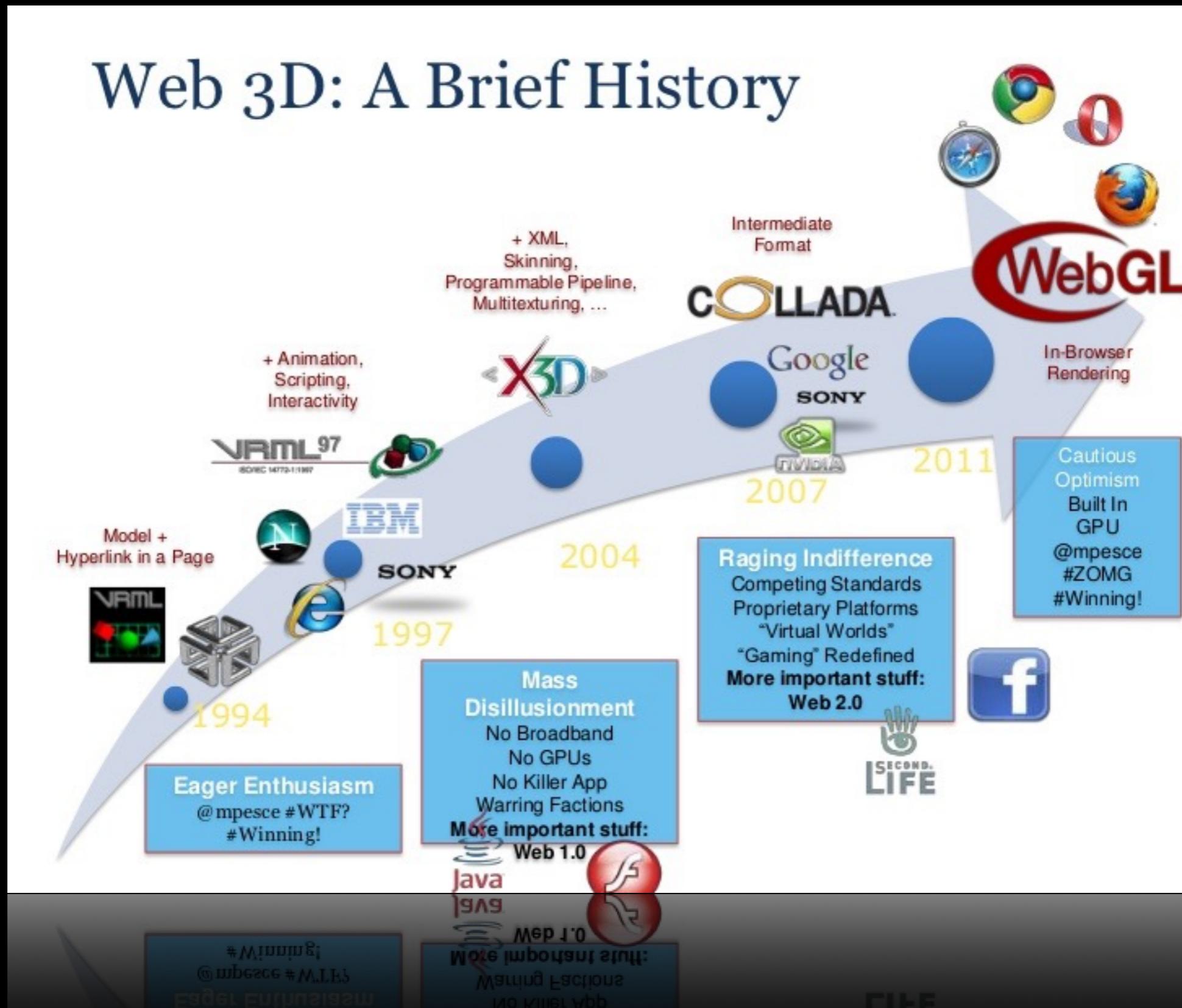
Some HTML5 Features...

- **Presentation** - CSS3/3D, Canvas (2D/3D*), Improved Document Semantics
- **Logic** - Fast JS
- **Data Persistence** - Local Storage, SQLite
- **Media** - Better Audio/Video Support
- **P2P Communication** - WebRTC
- **Networking** - Web Sockets
- **Multitasking** - Web Workers
- **Input Support** - Gamepad control, access to Web Cam

DEFINING WEBGL

- Low-level JavaScript 3D Graphics API
- Interfaces with OpenGL Graphics API Library.
- WebGL 1.0 - Based on OpenGL ES 2.0
- Uses OpenGL Shading Language (GLSL ES) which is compiled and runs on your computer's graphics processor (GPU)
- Piggybacks on the HTML5 Canvas Element

EVOLUTION OF GRAPHICS WEB SUPPORT



WEBGL ADVANTAGES

- It's a royalty-free, cross-platform 3D hardware accelerated API you can use without browser plug-ins.
- Based on the OpenGL ES 2.0 Standard - it is included in most modern browsers, including mobile devices.
- Can leverage all browser functionality.
- Doesn't require as much development tooling - you just need a text editor and a WebGL-enabled browser.
- The interfacing language is JavaScript!

WEBGL DISADVANTAGES

- It's low-level API and can be hard to learn/understand without the use of 3rd party libraries such as three.js.
- Does not take accessibility into account (it's developer dependent). Note that this is a problem with the HTML5 Canvas in general.
- Limited by the current features of the browser you are running your application on.
- Can be hard to debug.
- A potential security risk (potentially a DoS) because the shader programs written in OpenGL ES Shading Language needs to be compiled by the browser and run directly on the graphics hardware.
- Not compatible with older versions of browsers as well as some relatively recent ones (i.e. pre-Internet Explorer 11).

WEBGL SUPPORT

- **Mobile:**

- ✓ iOS 8+
- ✓ Android 4+
- ✓ BlackBerry 10+
- ✓ Windows Phone 11+

- **Desktop**

- ✓ Chrome
- ✓ Firefox
- ✓ Safari
- ✓ Opera
- ✓ IE 11+

WEBGL APPLICATIONS



GAMES - [HTTP://LABS.GOOENGINE.COM/PEARL-BOY/](http://LABS.GOOENGINE.COM/PEARL-BOY/)

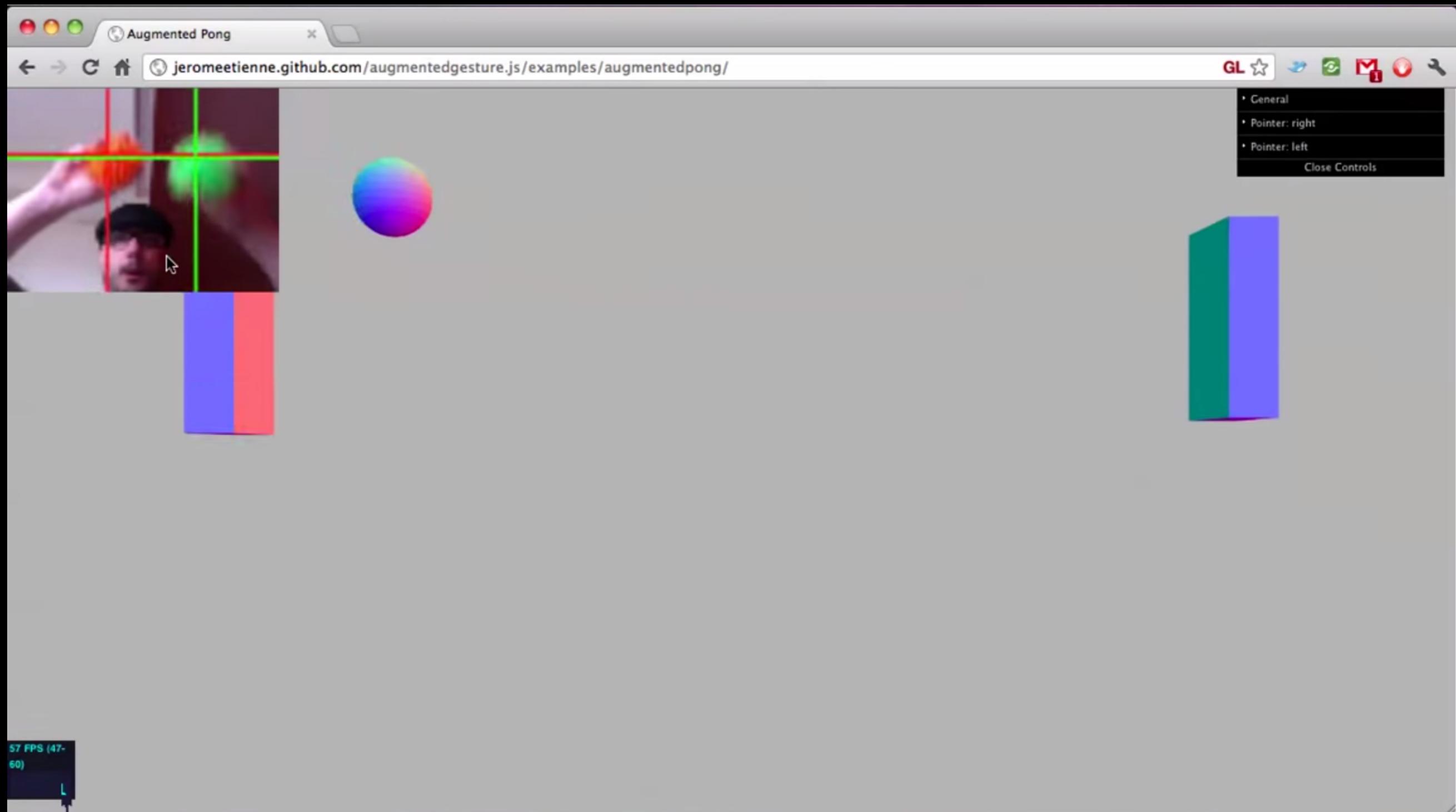


Pearl Boy was created with [Goo Engine](#), a 3D engine based on HTML5/WebGL.
[Goo Create](#) is a visual editor for making 3D apps and games and publishing them on the web.
[Click here to get started!](#)

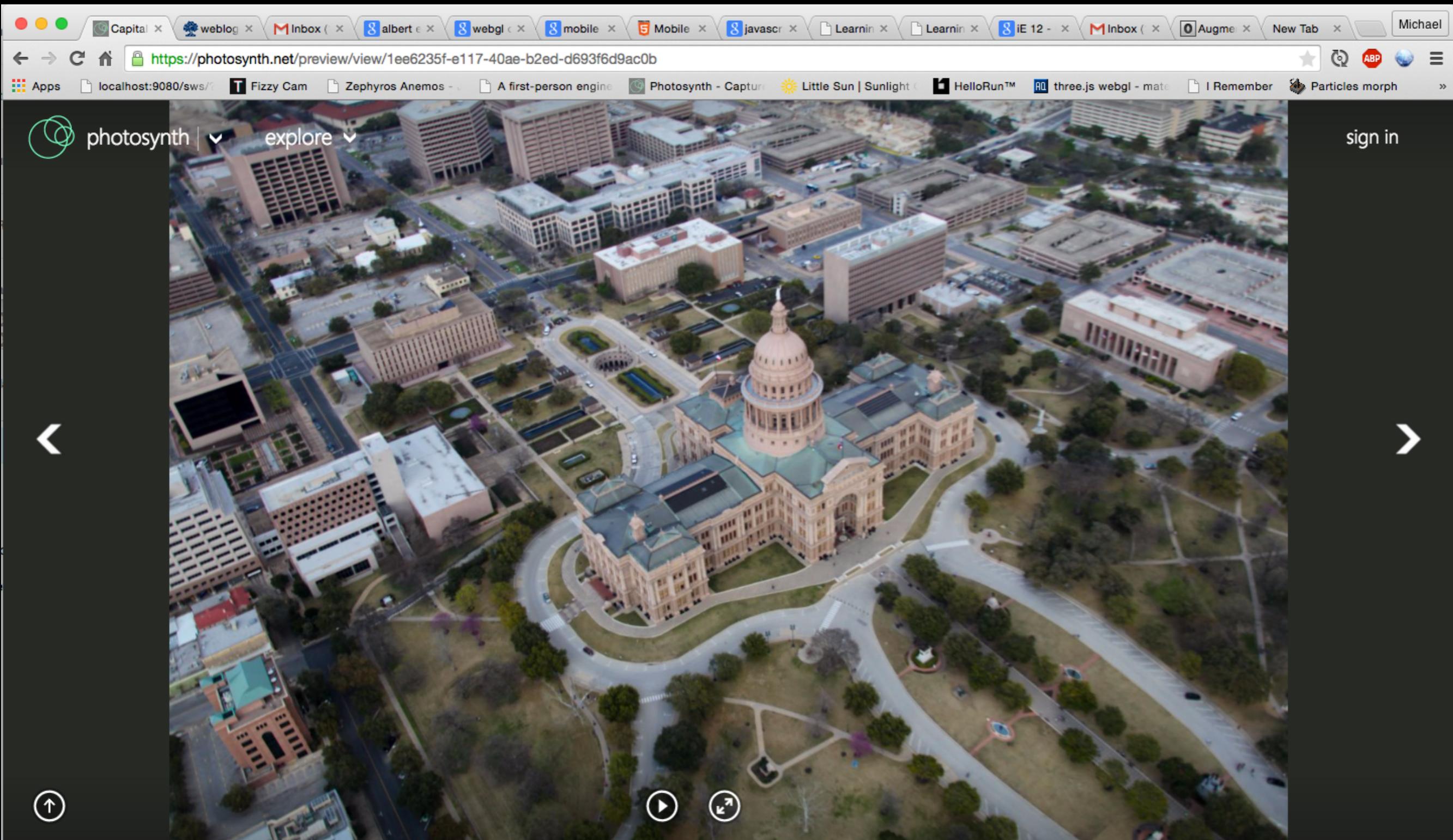
Powered by 



AUGMENTED REALITY - [HTTP://LEARNINGTHREEJS.COM/BLOG/2012/05/02/AUGMENTED-REALITY-3D-PONG/](http://learningthreejs.com/blog/2012/05/02/augmented-reality-3d-pong/)



SIMULATIONS - HTTPS://PHOTOSYNTH.NET/



ART - [HTTP://LIGHTGRAFFITI.LITTLESUN.COM/](http://lightgraffiti.littlesun.com/)

Little S × weblog × M Inbox (× g albert e × g webgl × g mobile × S Mobile × g javasc × Learnin × Learnin × g IE 12 - × M Inbox (× O Augme × New Tab × Michael

lightgraffiti.littlesun.com

Apps localhost:9080/sws/ Fizzy Cam Zephyros Anemos - A first-person engine Photosynth - Capture Little Sun | Sunlight HelloRun™ three.js webgl - mate I Remember Particles morph

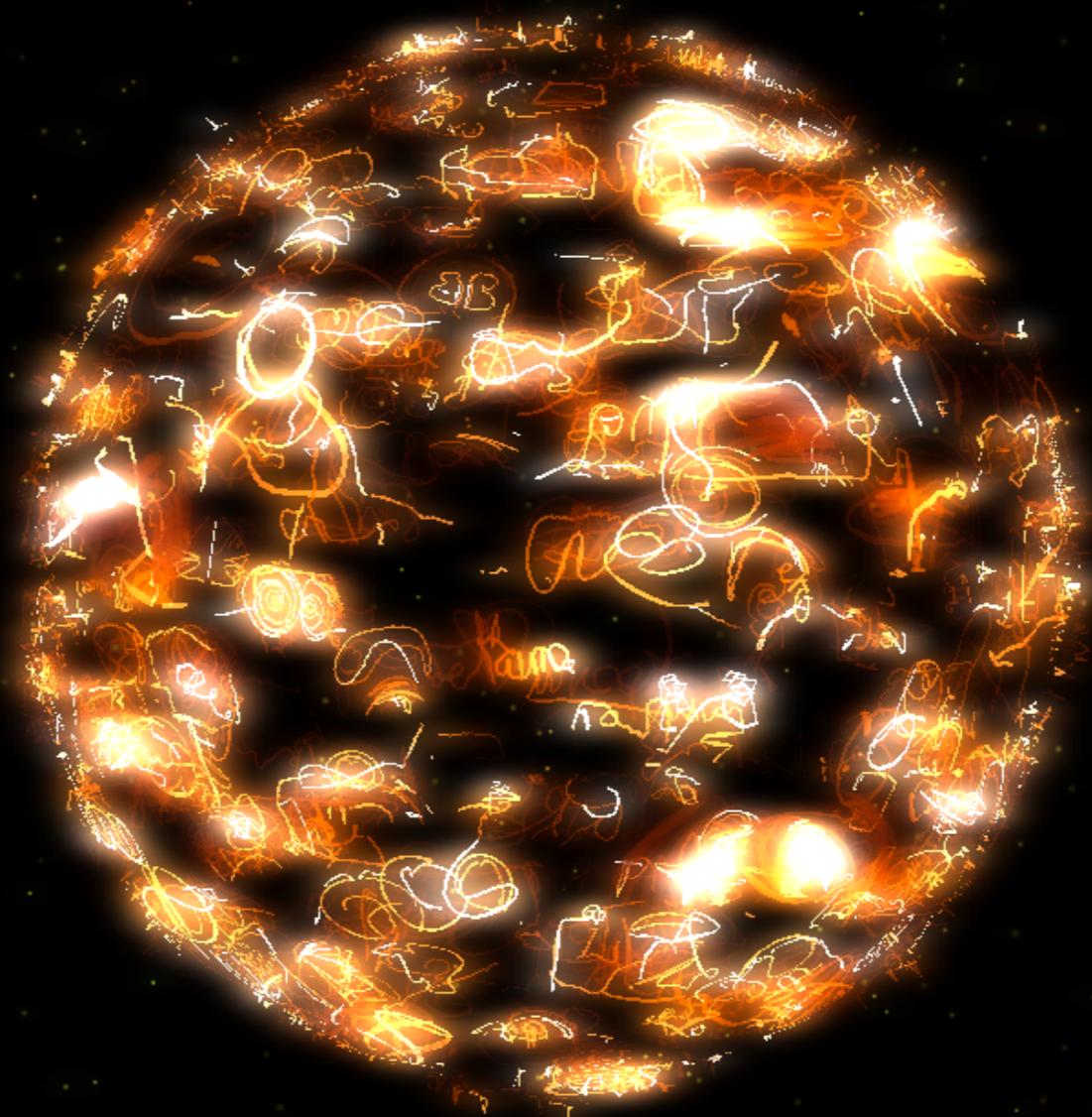
Little Sun Sunlight Graffiti

about | explore | credits | watch video

Did you participate at Tate Modern?
Find your graffiti

10949 entries

Showing 10779 – 10949



A project by Olafur Eliasson at Tate Modern littlesun.com

[Twitter 287](#) [Facebook Like 998](#)

VISUALIZATIONS - [HTTP://I-REMEMBER.FR/EN](http://i-remember.fr/en)

i-remember.fr/en

Let's share our memories
and make sure this website does not disappear.

14%
FADE FROM
MEMORIES

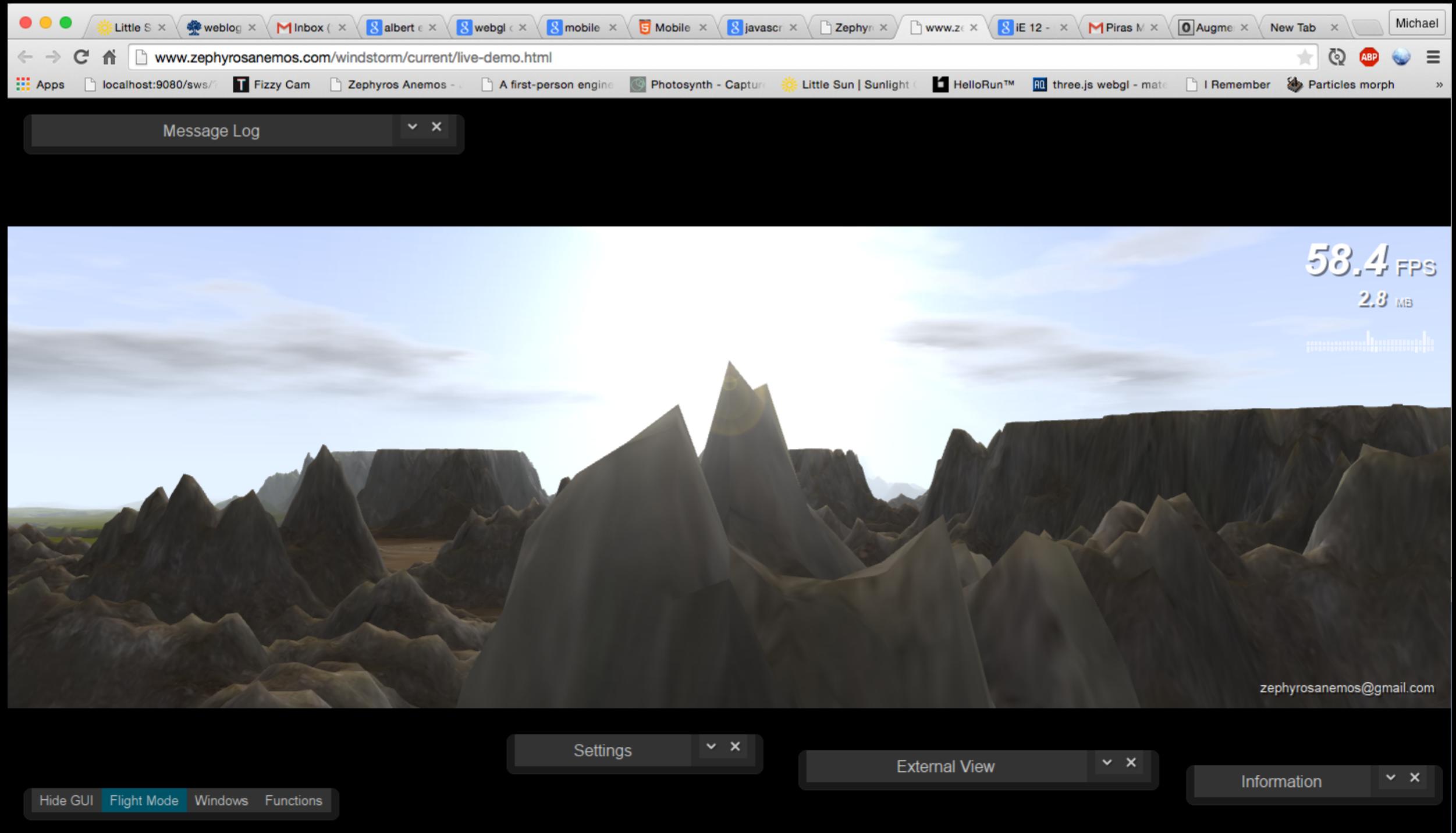
+ tell a
memory

see all memories

f t g+ D O N A T E T E R M S A N D C O N D I T I O N S C R E D I T S L A N G U E _ E N ▾

FONDATION
RECHERCHE
MÉDICALE

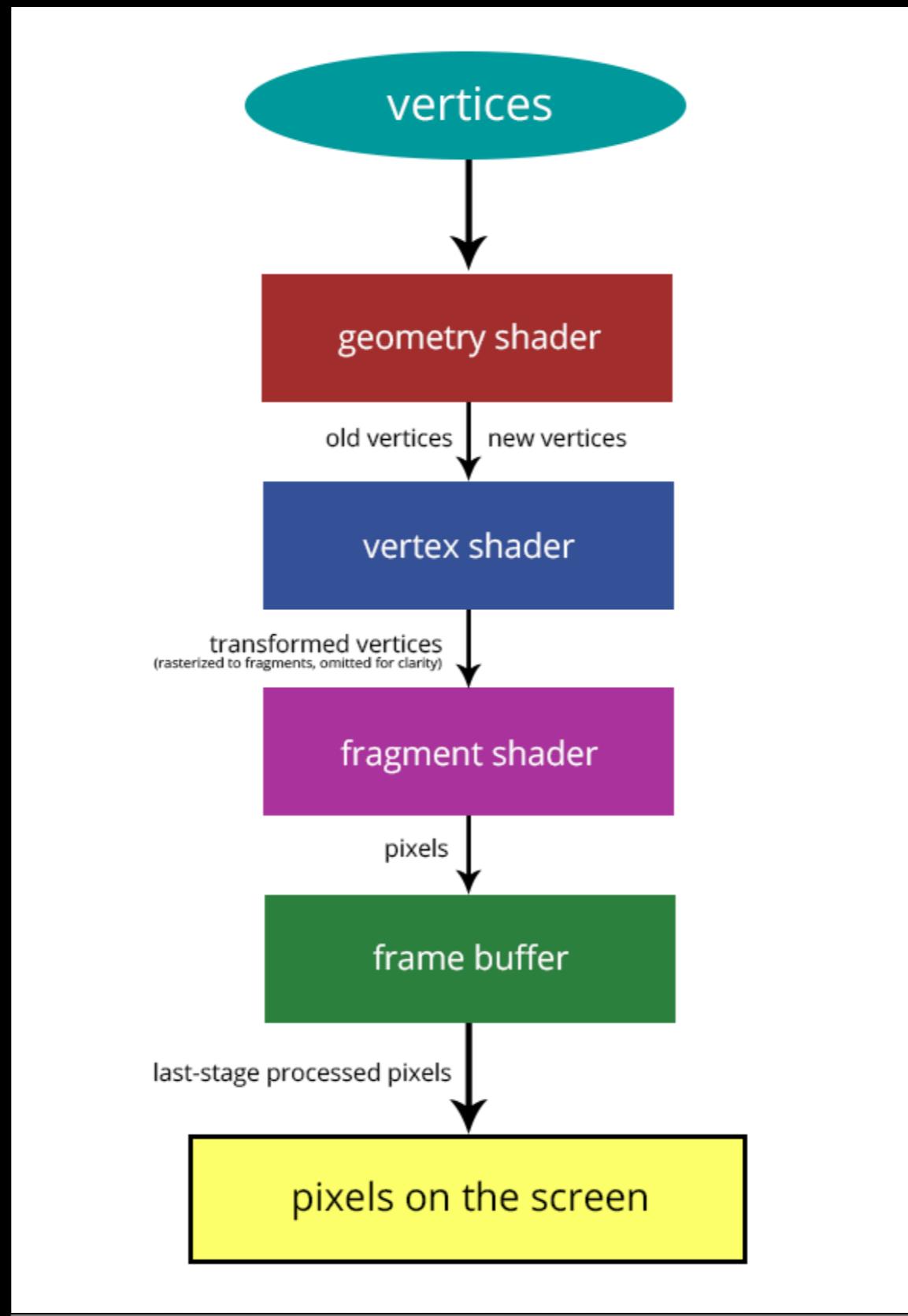
GRAPHICS ENGINES - HTTP:// WWW.ZEPHYROSANEMOS.COM/WINDSTORM/CURRENT/ LIVE-DEMO.HTML



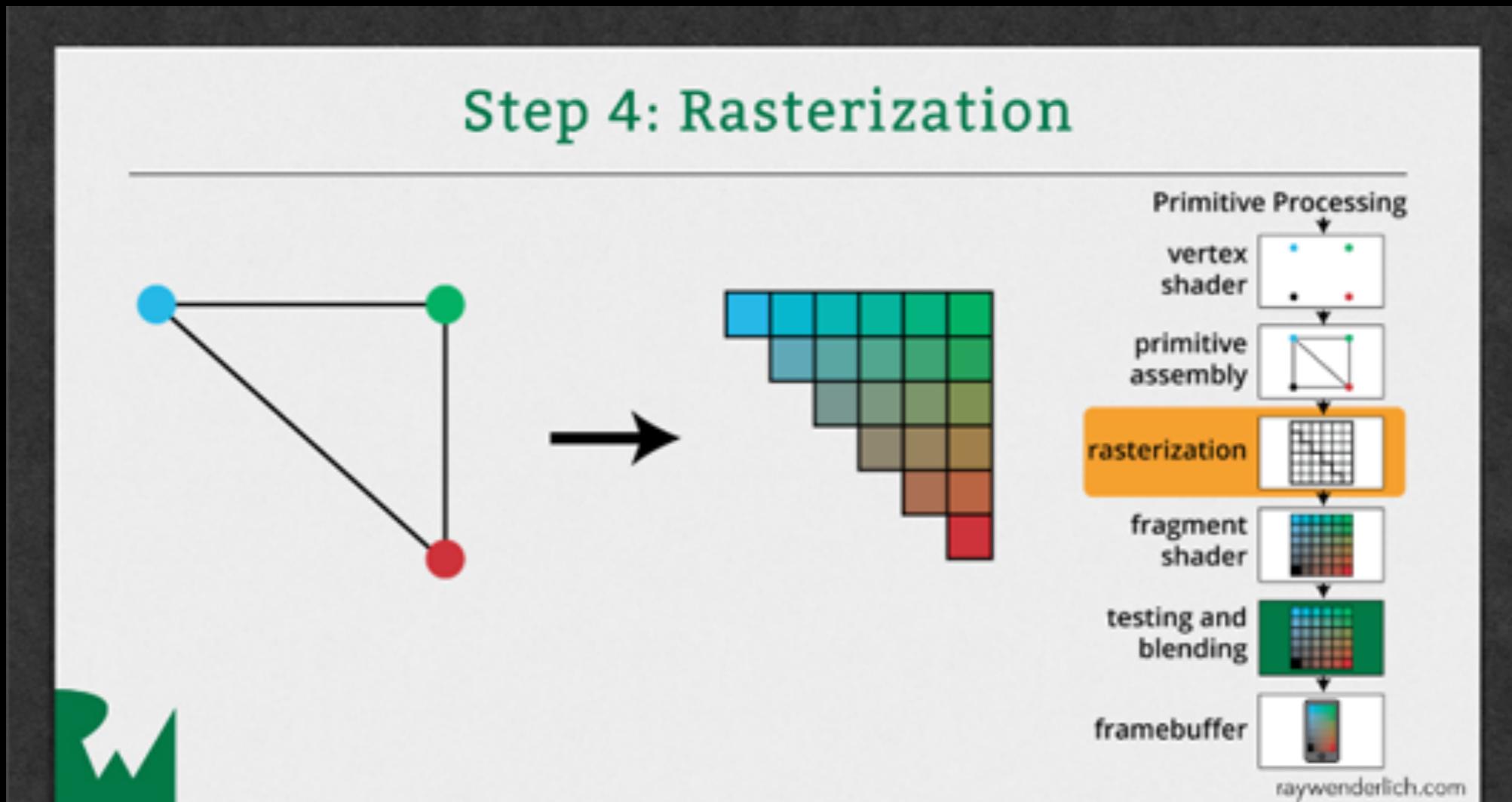
SOME (OTHER) WEBGL APPLICATIONS

- Water Simulation: <https://www.shadertoy.com/view/Ms2SD1>
- 3D Simulation: <http://labs.gooengine.com/pearl-boy/>
- Movement Detection with Camera Support - <http://www.clicktorelease.com/code/optical-flow-webrtc/>
- http://learningwebgl.com/blog/?page_id=1217

WEBGL RENDERING PIPELINE



ANOTHER EXAMPLE OF THE PIPELINE



raywenderlich.com
tutorials

A SAMPLE VERTEX SHADER

```
<script id="v-shader-demo4" type="x-shader/x-vertex">

// attributes are accessible from JavaScript (but are write only - since we do most of our work
in Javascript it doesn't matter)
attribute vec4 a_Position;
attribute vec4 a_Colour;
varying vec4 v_Colour; // this represents the colour value we want to interpolate by
(between each vertex)

void main() {
    gl_Position = a_Position; // specify a custom vertex from JavaScript
    gl_PointSize = 1.0; // each pixel will be 1:1

    // giving this variable same name as the variable in the fragment shader effectively passes
    // on the argument
    v_Colour = a_Colour;
}

</script>
```

A SAMPLE FRAGMENT SHADER

```
<script id="f-shader-demo4" type="x-shader/x-fragment">
/*
    New comment multi-line style shown here -
    because we specified the same variable name in the
*/
varying mediump vec4 v_Colour;

void main() {
    gl_FragColor = v_Colour; // assign the interpolated colour
}
</script>
```

HOW TO GET STARTED

- Learn JS
- Learn the WebGL API
- Learn GLSL ES



DEBUGGING WEBGL

- Primary Resource - <https://www.khronos.org/webgl/wiki/Debugging>
- General Debugging - <https://www.khronos.org/registry/webgl/sdk/debug/webgl-debug.js>
- Interactive Debugging - <http://benvanik.github.io/WebGL-Inspector/>

START LEARNING HERE...

- <http://learn.npcomplete.me/webgl/>
- <https://github.com/mikecules/WebGL-RLQ>

RESOURCES

- ShaderToy - <https://www.shadertoy.com/>
- OpenGL ES SL Manual - https://www.khronos.org/files/opengles_shading_language.pdf
- OpenGL ES Ref Card - https://www.khronos.org/opengles/sdk/docs/reference_cards/OpenGL-ES-2_0-Reference-card.pdf
- three.js - <http://threejs.org/>
- Tutorials - <http://www.raywenderlich.com/tag/opengl>

Michael Moncada

MOBILE APPLICATION DEVELOPER

MICHAEL.MONCADA@UTORONTO.CA