



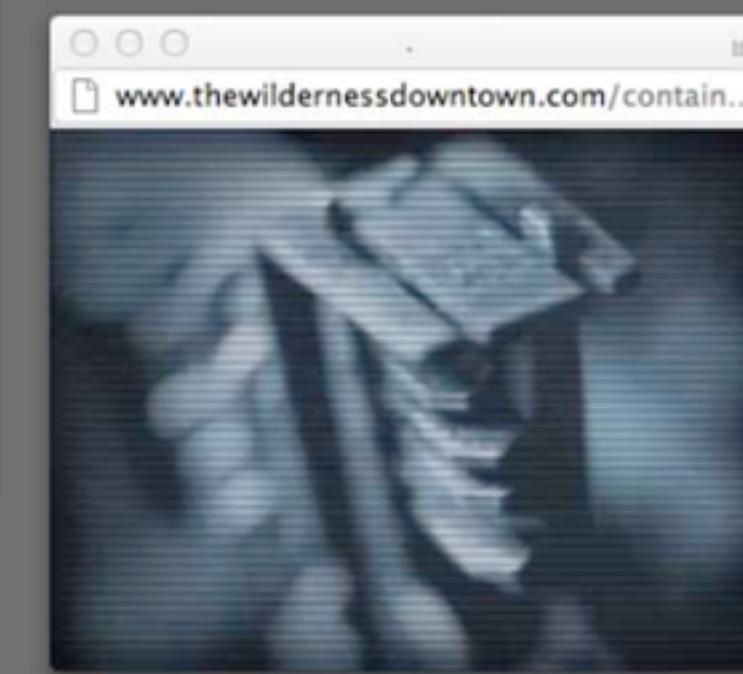
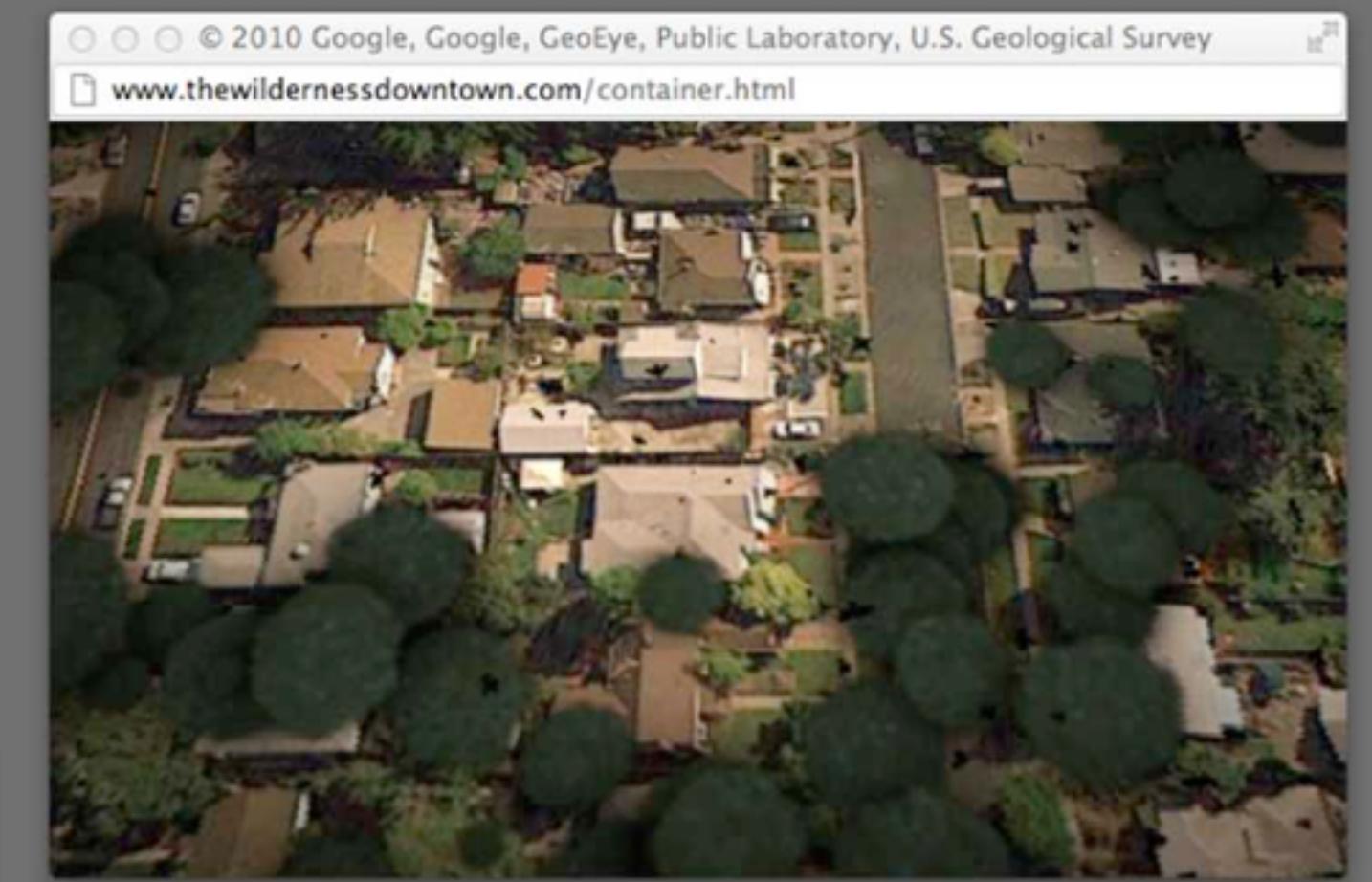
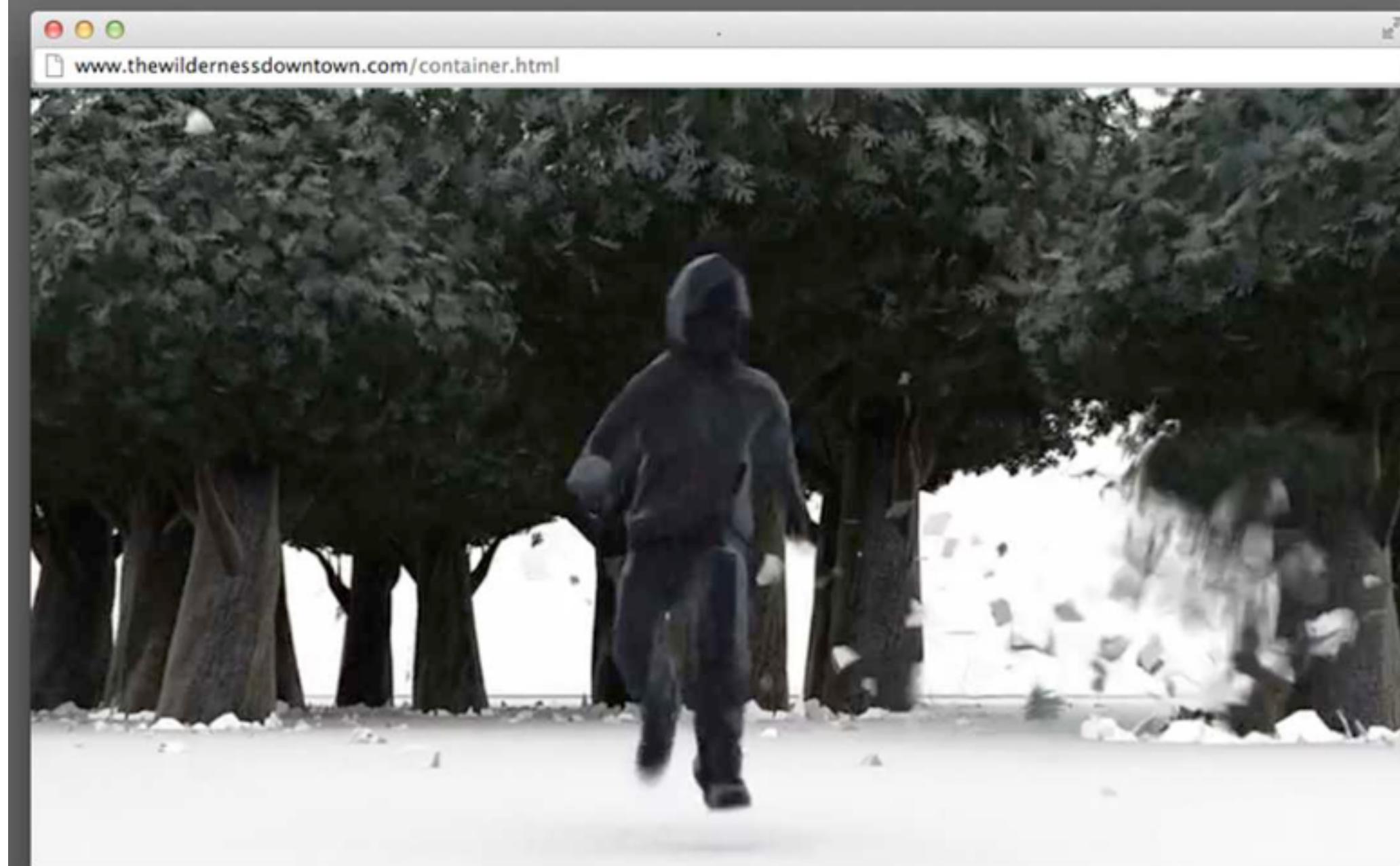
# Getting started with WebGL and three.js

The Graphical Web 2014

Jaume Sanchez Elias

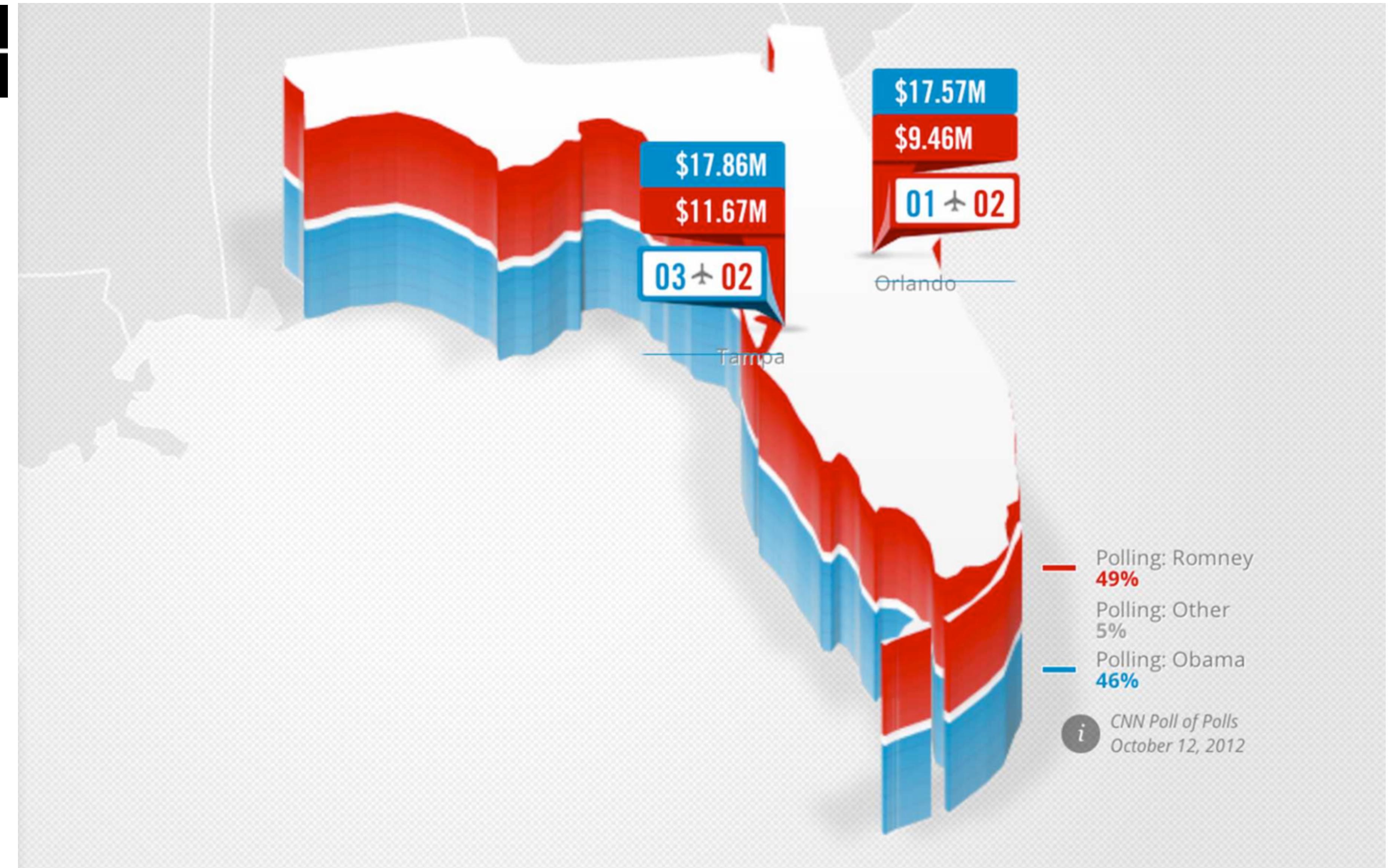
<http://www.clicktorelease.com>

@thespite



The Wilderness Downtown | Chrome Experiment | CSS + Canvas

B REEL



Google/CNN Campaign Tracker | SVG + Canvas

BREE

CUBE

A game about Google maps

Watch the Explore your world video

Visit Start here



Level 4/8: Find the fastest subway route to the Big Ben!

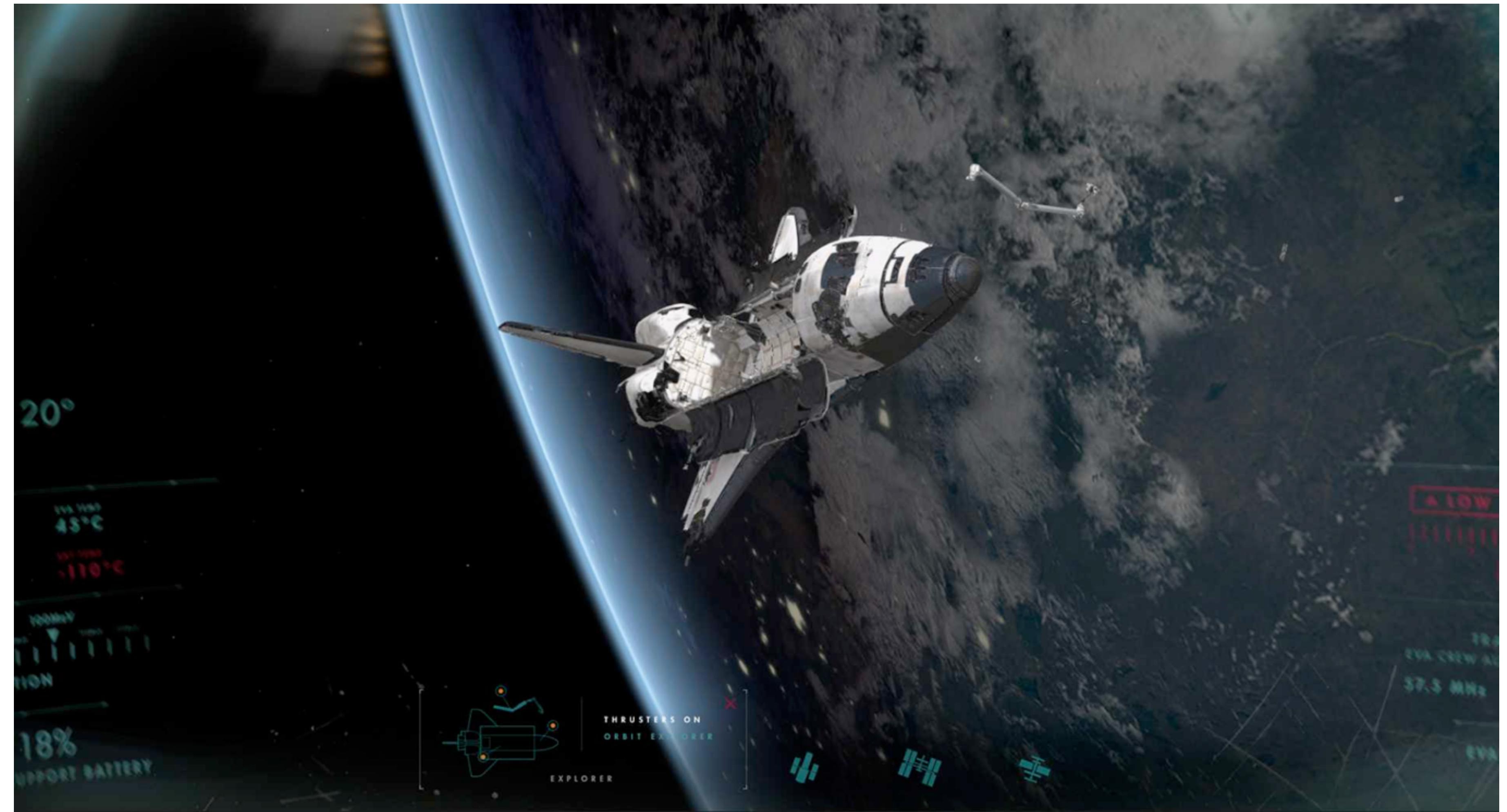
Your Time: 00:02

Best: 00:06

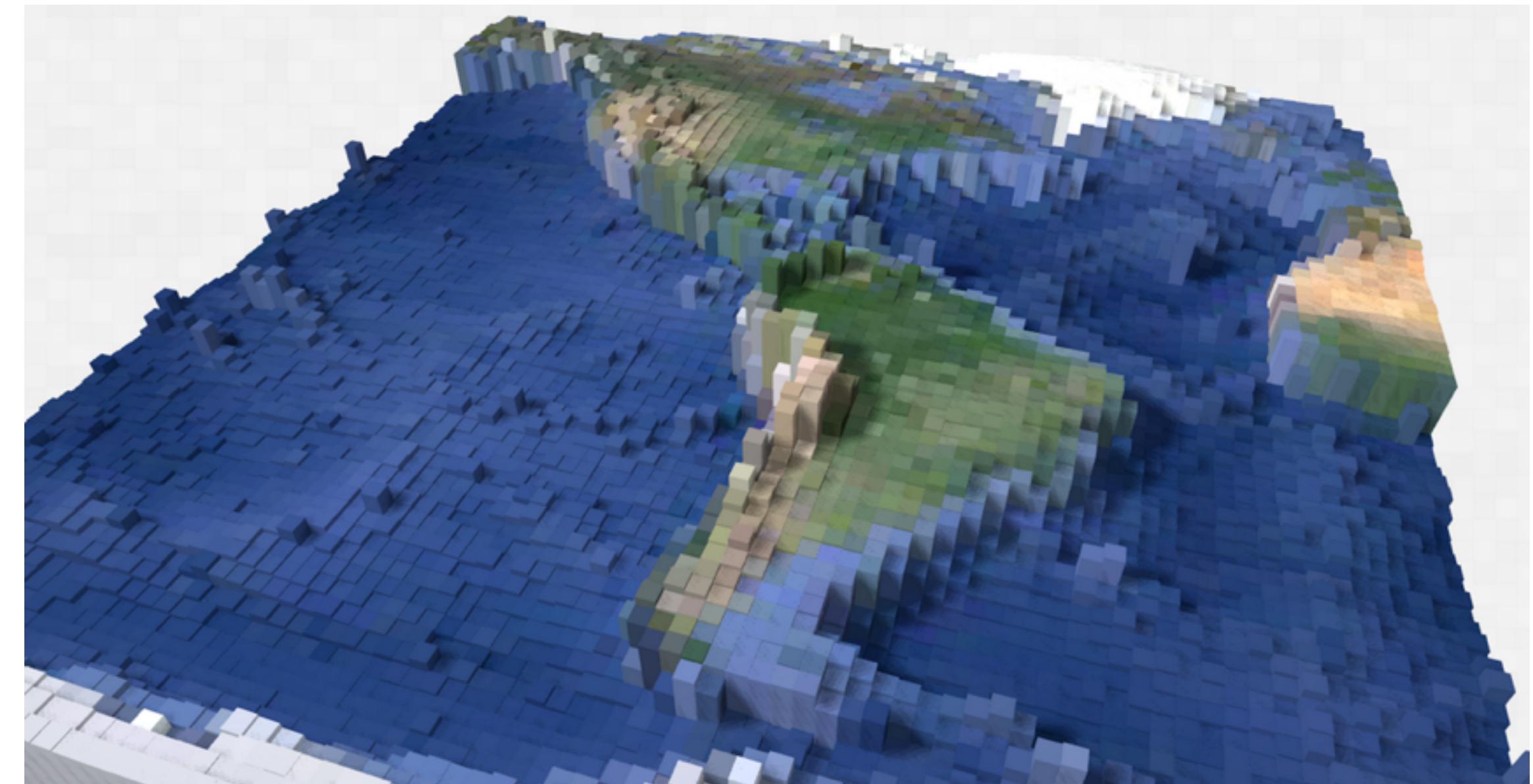
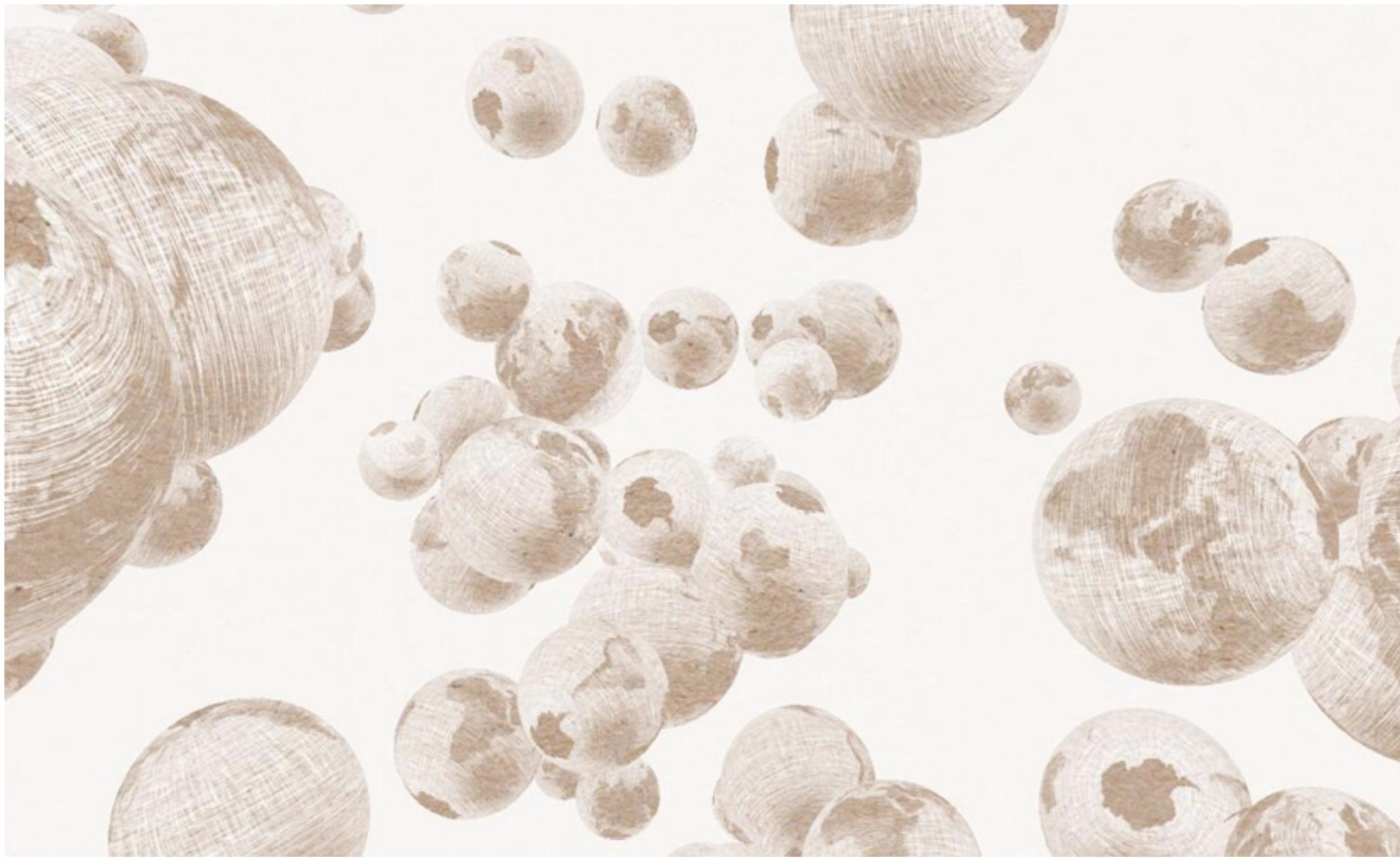
Restart

Google Maps Cube Game | Chrome Experiment | WebGL

BREEEL



Gravity | Official movie website | WebGL



Surfacing, shading, post-processing

# WebGL

"WebGL (Web Graphics Library) is a JavaScript API for rendering interactive 3D graphics and 2D graphics within any compatible web browser without the use of plug-ins."

<http://www.khronos.org/webgl/>

# Fundamentals of WebGL

## GPU overview

A graphics processing unit (GPU) is a specialised electronic circuit designed to rapidly manipulate and alter memory to accelerate the creation of images in a frame buffer intended for output to a display.

Modern GPUs are very efficient at manipulating computer graphics, and their highly parallel structure makes them more effective than general-purpose CPUs for algorithms where processing of large blocks of data is done in parallel.

# Fundamentals of WebGL

## From OpenGL to WebGL

WebGL is a cross-platform, royalty-free web standard for a low-level 3D graphics API based on OpenGL ES 2.0.

It's exposed through the HTML5 Canvas.

Very close to the OpenGL ES 2.0 specification with some concessions made for JavaScript.

# **three.js**

"The aim of the project is to create a lightweight 3D library with a very low level of complexity – in other words, for dummies.

The library provides <canvas>, <svg>, CSS3D and WebGL renderers."

<http://threejs.org/>

## Setting up

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

var scene = new THREE.Scene();
var aspectRatio = window.innerWidth / window.innerHeight;
var camera = new THREE.PerspectiveCamera( 75, aspectRatio, 0.1, 1000 );

var renderer = new THREE.WebGLRenderer();
renderer.setSize( window.innerWidth, window.innerHeight );
document.body.appendChild( renderer.domElement );

var geometry = new THREE.BoxGeometry(1,1,1);
var material = new THREE.MeshBasicMaterial( { color: 0x00ff00 } );
var cube = new THREE.Mesh( geometry, material );
scene.add( cube );

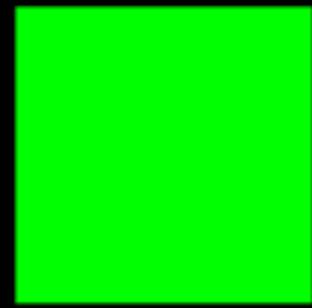
camera.position.z = 5;

function render() {
    requestAnimationFrame(render);
    renderer.render(scene, camera);
}

render();

</script>
```

Congratulations! It's a... cube?



# Geometries

BoxGeometry  
CircleGeometry  
CubeGeometry  
CylinderGeometry  
ExtrudeGeometry  
IcosahedronGeometry  
LatheGeometry  
OctahedronGeometry  
ParametricGeometry  
PlaneGeometry  
PolyhedronGeometry  
RingGeometry  
ShapeGeometry  
SphereGeometry  
TetrahedronGeometry  
TextGeometry  
TorusGeometry  
TorusKnotGeometry  
TubeGeometry

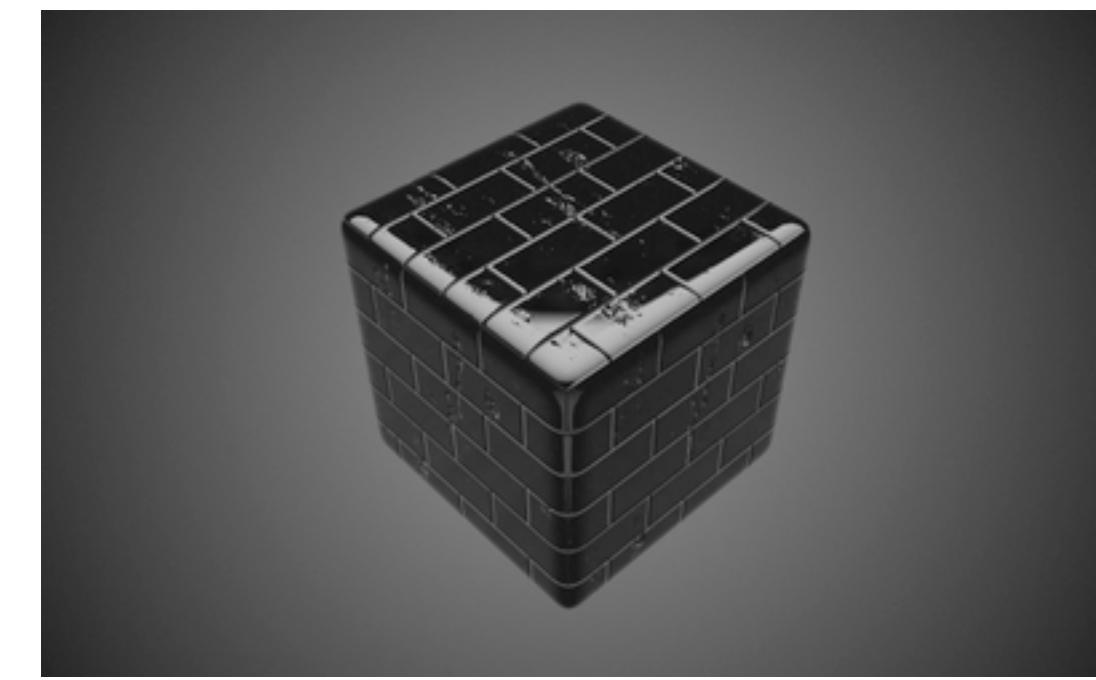
# Materials

LineBasicMaterial  
LineDashedMaterial  
MeshBasicMaterial  
MeshDepthMaterial  
MeshFaceMaterial  
MeshLambertMaterial  
MeshNormalMaterial  
MeshPhongMaterial  
PointCloudMaterial  
RawShaderMaterial  
ShaderMaterial  
SpriteCanvasMaterial  
SpriteMaterial

# Lights

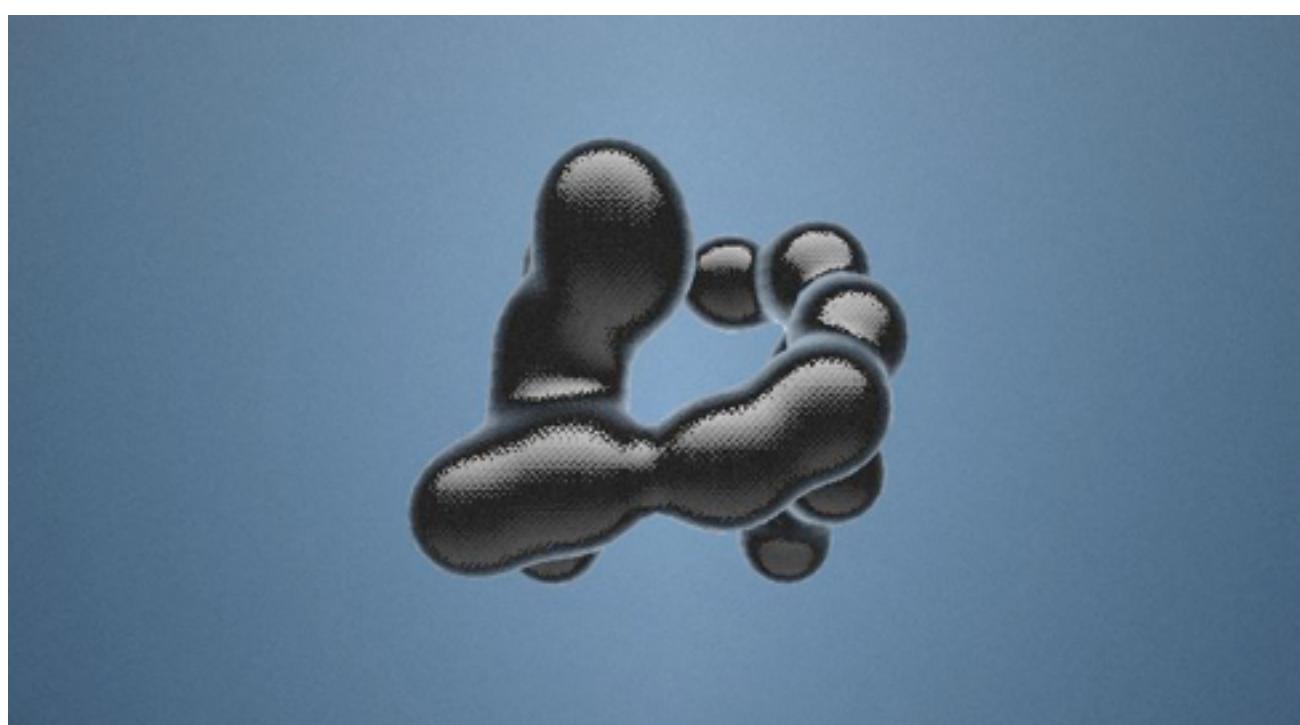
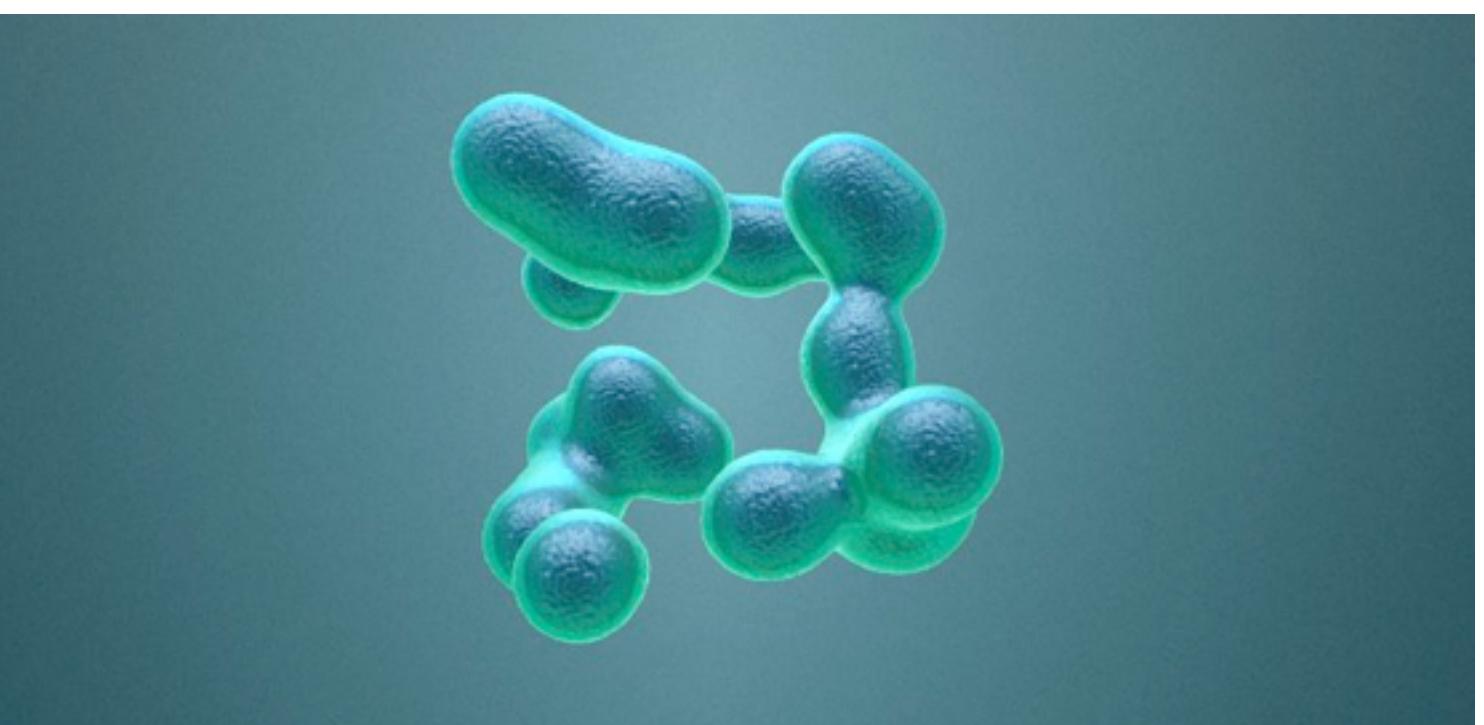
AmbientLight  
AreaLight  
DirectionalLight  
HemisphereLight  
PointLight  
SpotLight

# Setting the scene



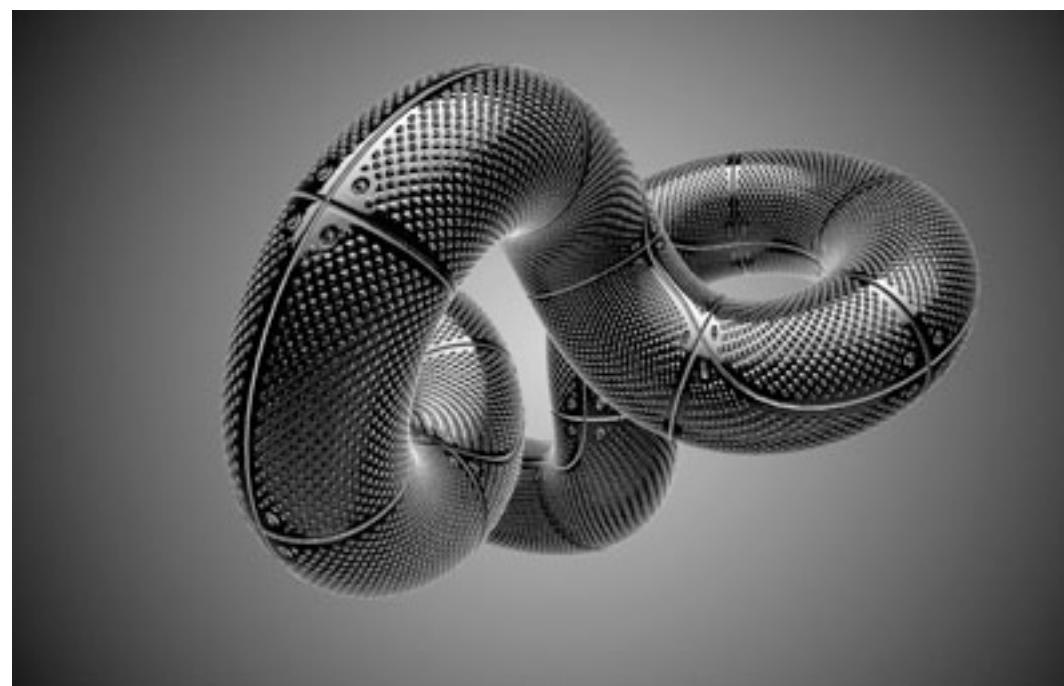
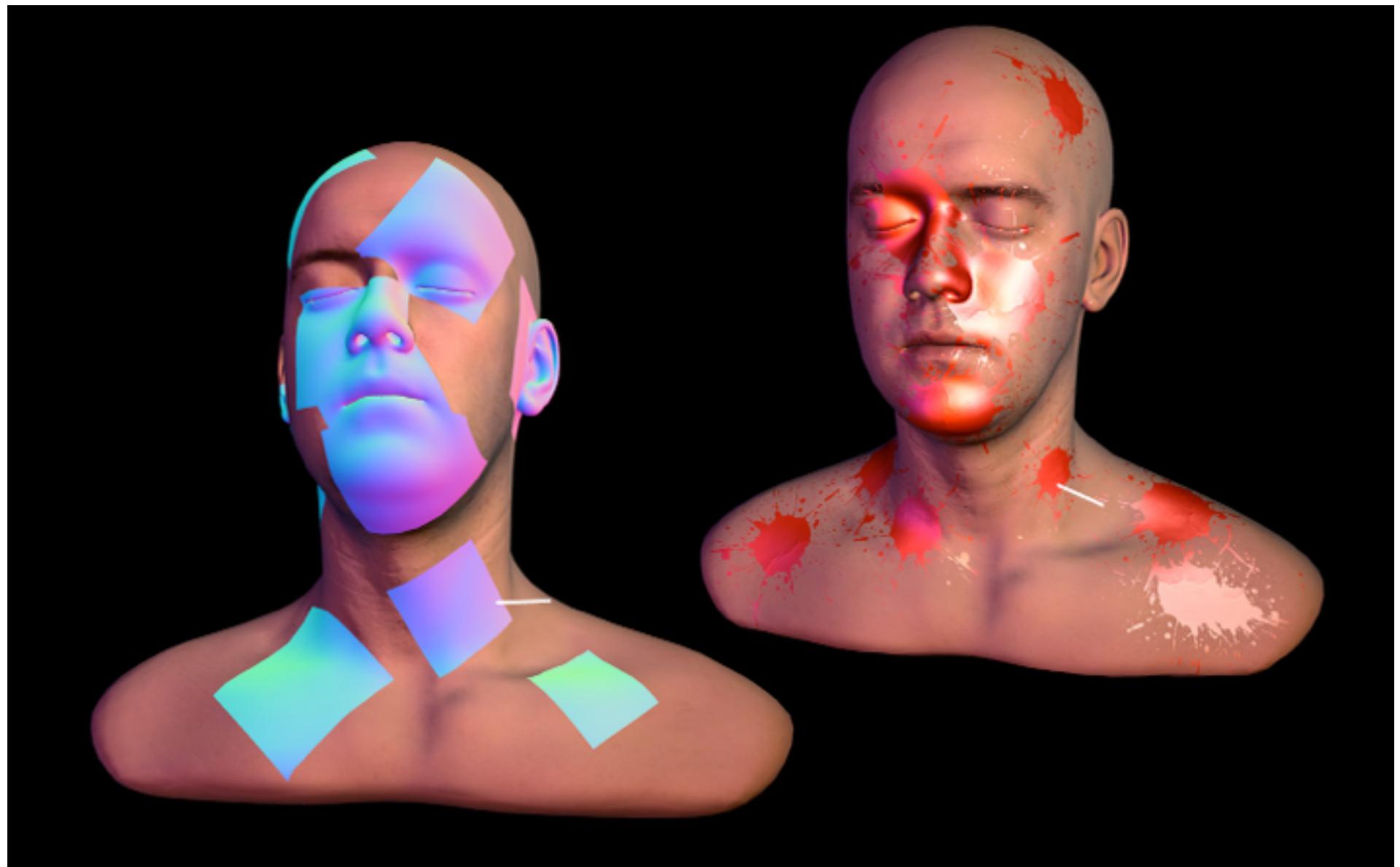
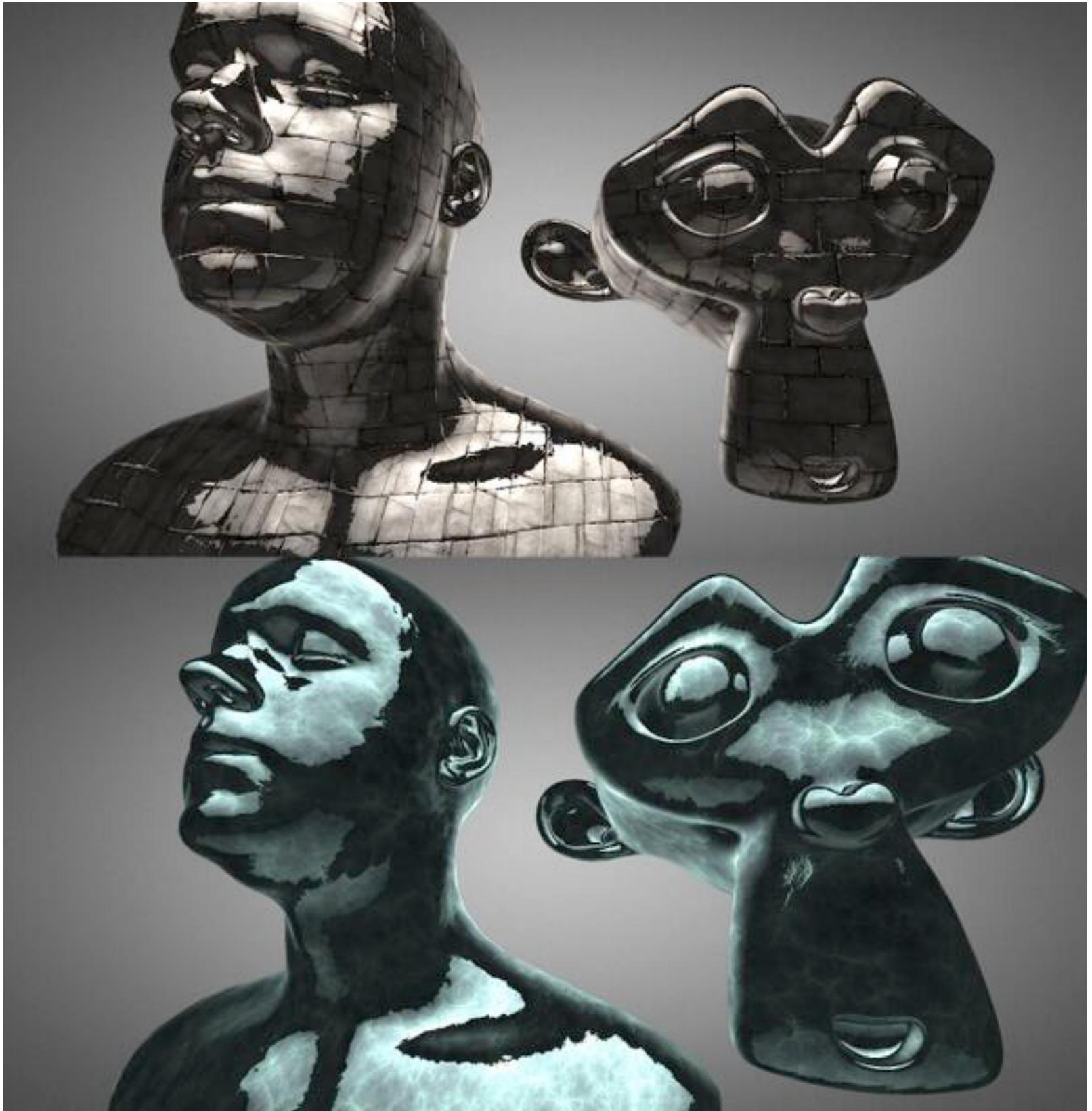
# Cameras

OrthographicCamera  
PerspectiveCamera

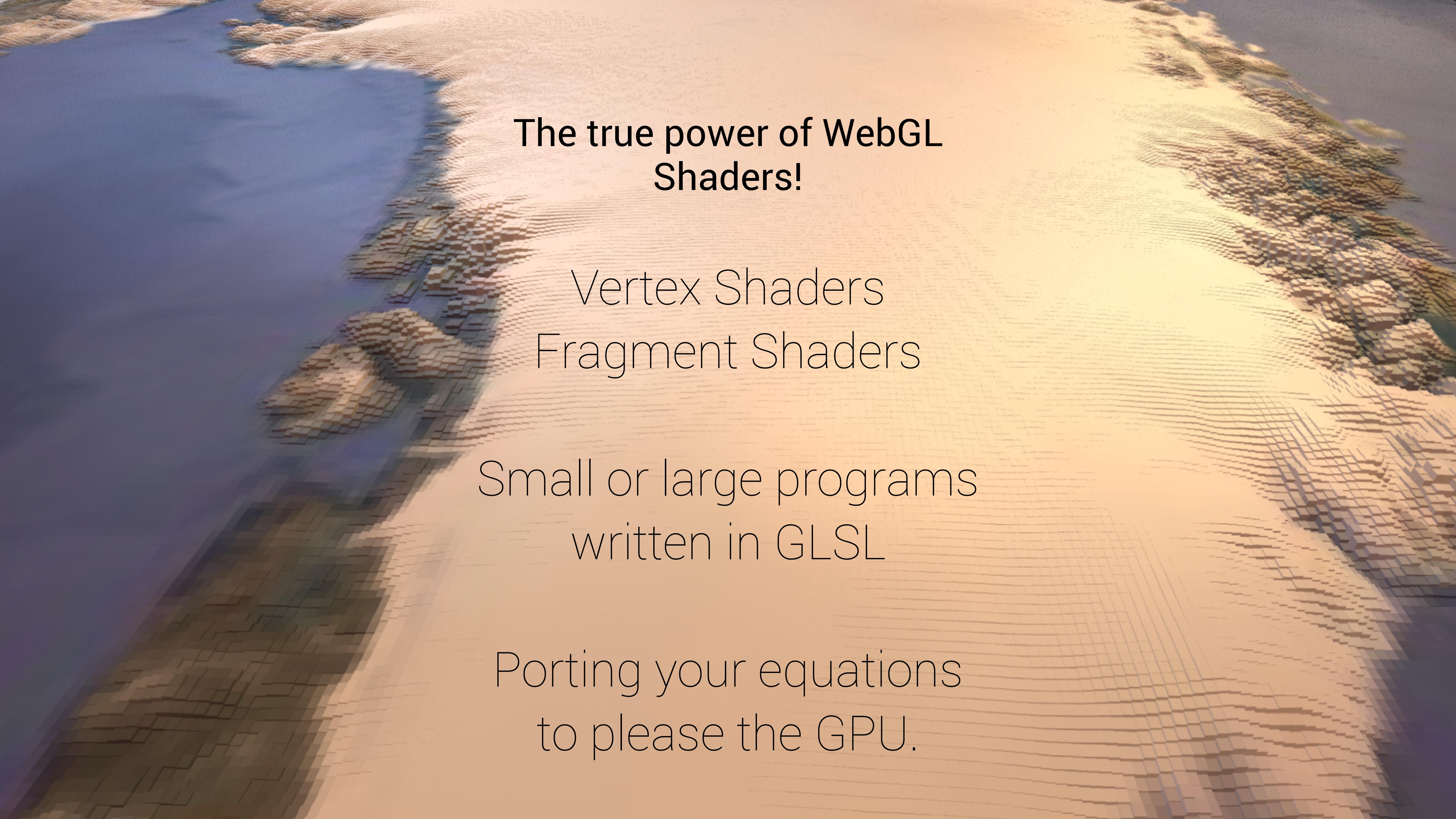


# Loaders

BufferGeometryLoader  
ImageLoader  
JSONLoader  
MaterialLoader  
ObjectLoader  
TextureLoader  
XHRLoader



**Enough cubes!**

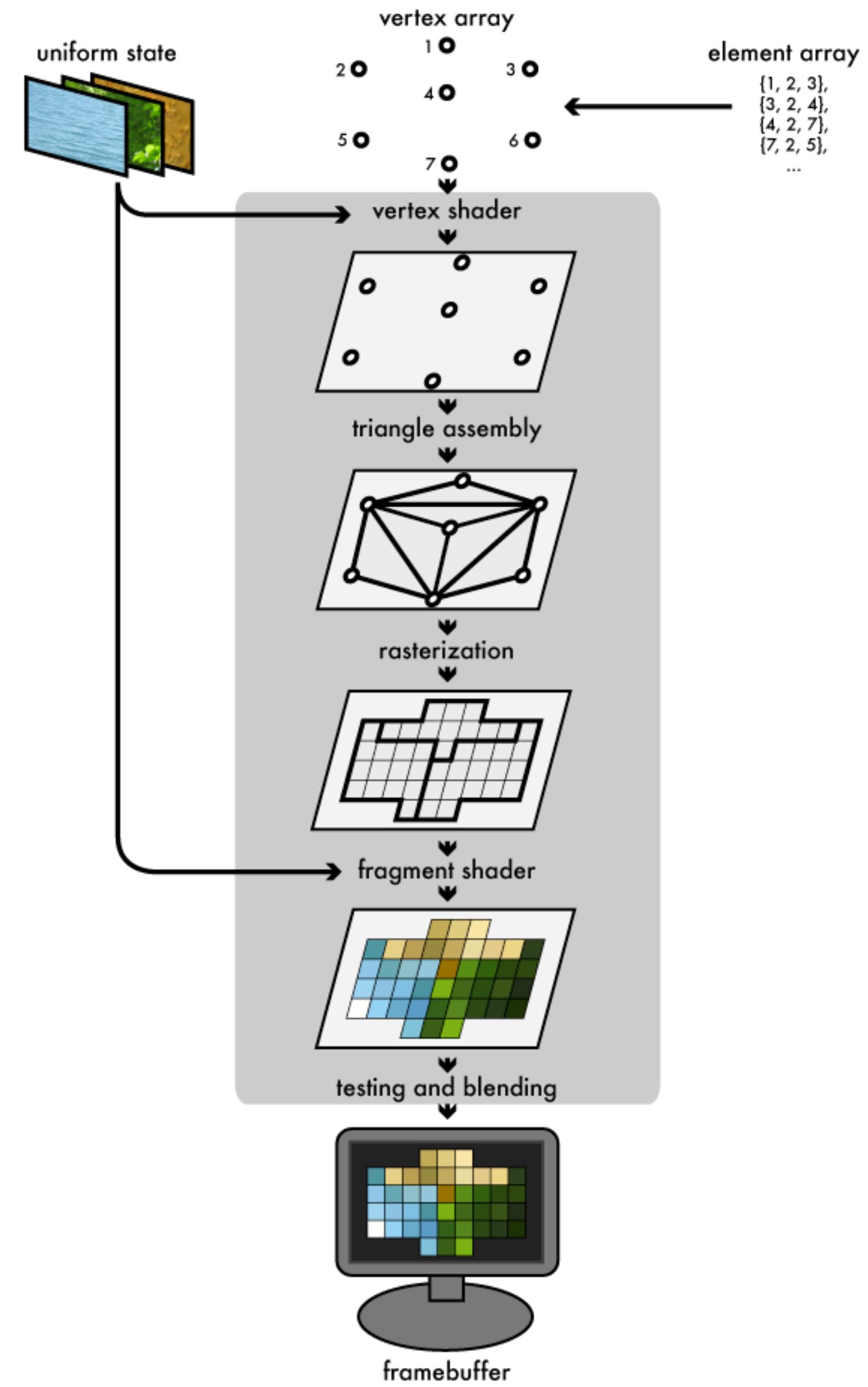


# The true power of WebGL Shaders!

Vertex Shaders  
Fragment Shaders

Small or large programs  
written in GLSL

Porting your equations  
to please the GPU.



# Thanks!

Questions?

Afraid of polygons?

Programmable pipelines make you queasy?

Let's talk!

Jaume Sanchez @thespite

<http://www.clicktorelease.com>

<https://github.com/spite>