

Building Blocks



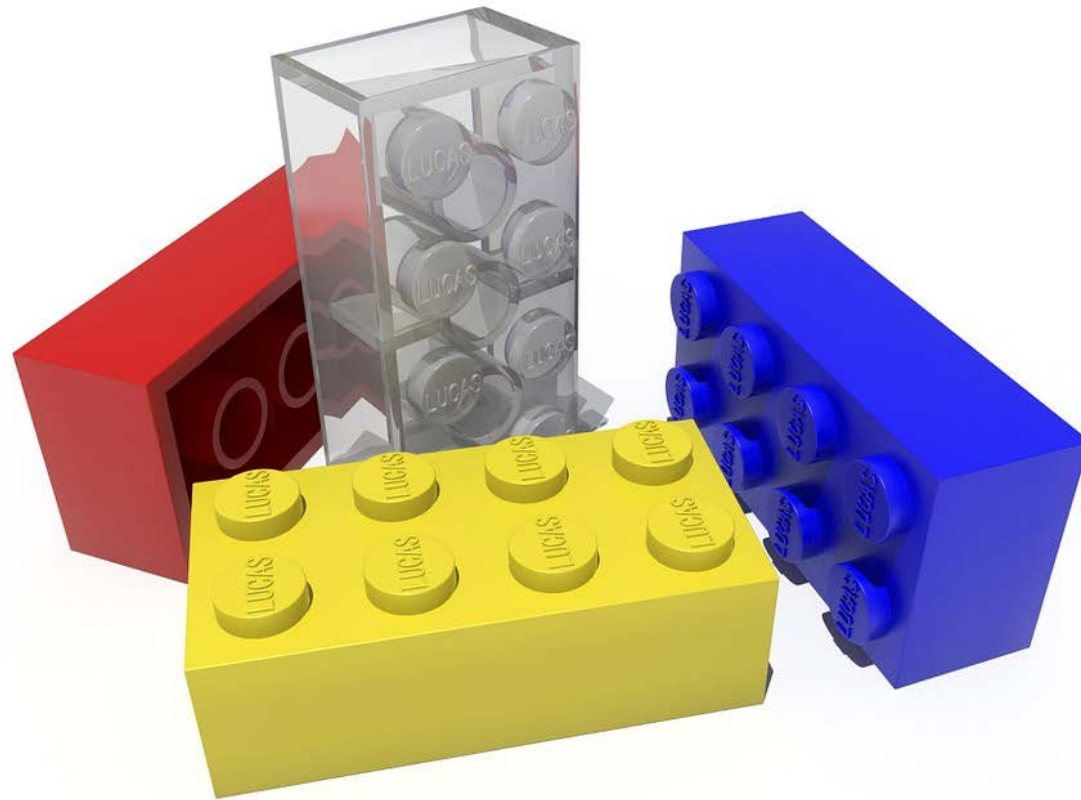
Alex Mackey

MVP - VISUAL STUDIO & DEVELOPMENT TECHNOLOGIES

@alexjmackey www.simpleisbest.co.uk



Building Blocks



Building Block Usages

Debugging

Extension

Other Usages



Building Blocks

JavaScript

WebVR

WebGL

Three.js

Device Orientation
Events



Other Technologies

Gamepad API

Web Audio API



WebVR



Warning Experimental!



WebVR Capabilities

**Obtain Device Capability +
Positioning Information**

Display Content Via WebGL



Browser Support



```
navigator.getVRDisplays()
```

```
if (!navigator.getVRDevices){  
    //no WebVR support  
}  
  
navigator.getVRDisplays().then(function (displays) {  
    if(displays.length===0){  
        //WebVR supported but no devices  
    }  
  
    vrDisplay = displays[0]; //VRDisplay object  
}
```



VRDisplay

```
vrDisplay.displayId
```

```
vrDisplay.getPose();
```

```
vrDisplay.getEyeParameters("left");
```

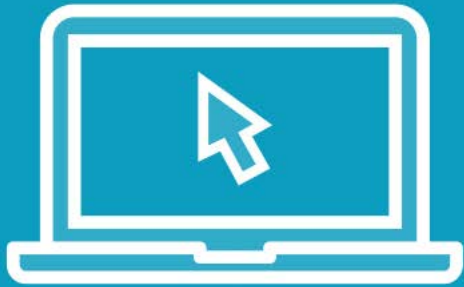
```
vrDisplay.getFrameData(frameData);
```

```
vrDisplay.requestAnimationFrame(onAnimationFrame);
```

```
vrDisplay.requestPresent(layers);
```



Demo



WebVR Basics



WebGL



WebGL Advantages

Good Support

**Use With Other
Web Technologies**

**Doesn't Require
Plugin**



WebGL Advantages

Good Support

**Use With Other
Web Technologies**

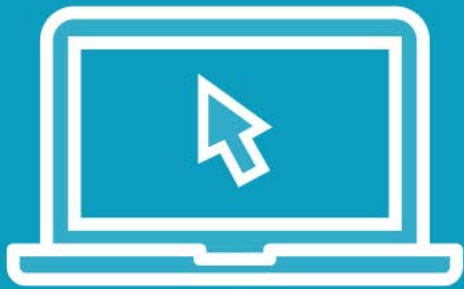
**Doesn't Require
Plugin**



Vladimir Vukićević



Demo



Hello WebGL!



Three.js



Three.js

Library created by Ricardo Cabello

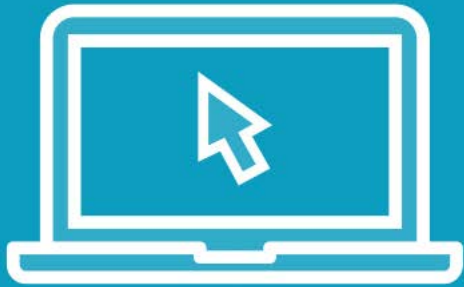
Background in demo scene



Three.js Concepts



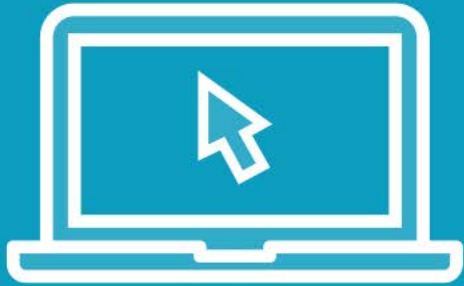
Demo



Hello Three.JS Demo



Demo



Three.JS & VR Demo



Device Orientation Events



Devices



No Control!



Device Orientation Events

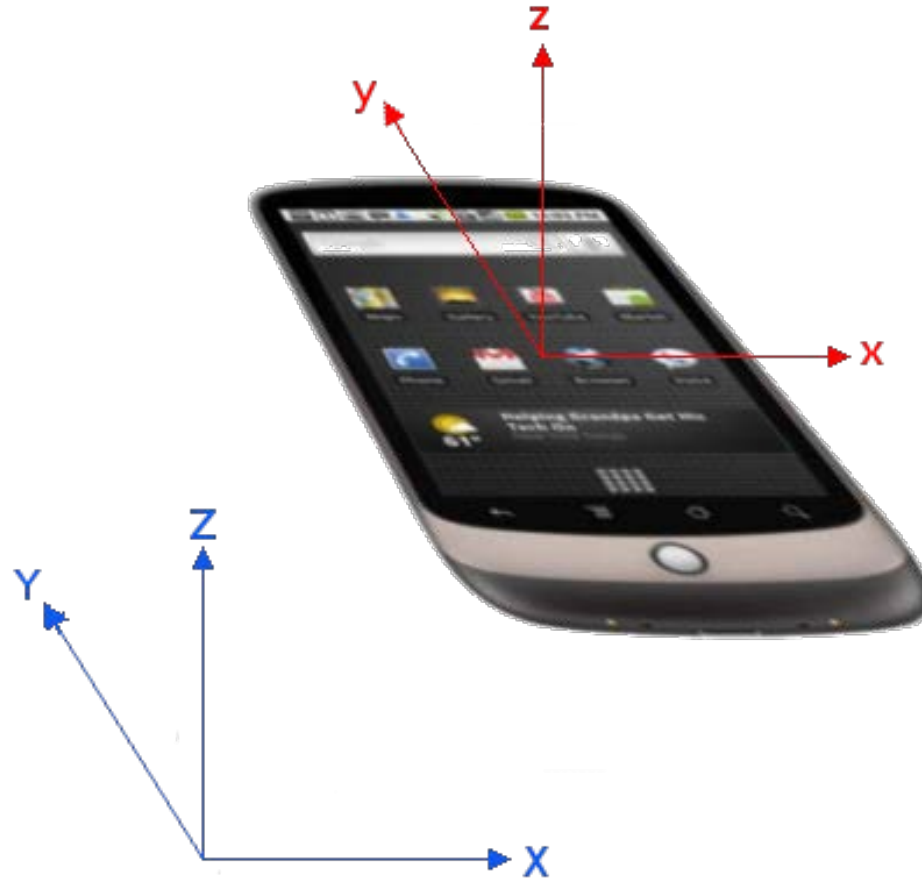
deviceorientation

devicemotion

**compassneedscale
ibration**



Device Orientation Axis



<https://www.w3.org/TR/2016/CR-orientation-event-20160818/>



deviceorientation

```
window.addEventListener  
("deviceorientation", handle, true);  
  
function handle(e) {  
    e.absolute;  
    e.alpha; //z axis 0 to 360  
    e.beta;  //x axis -180 to 180  
    e.gamma; //y axis -90 to 90  
}
```



Alpha (Z)
“Rotating phone”

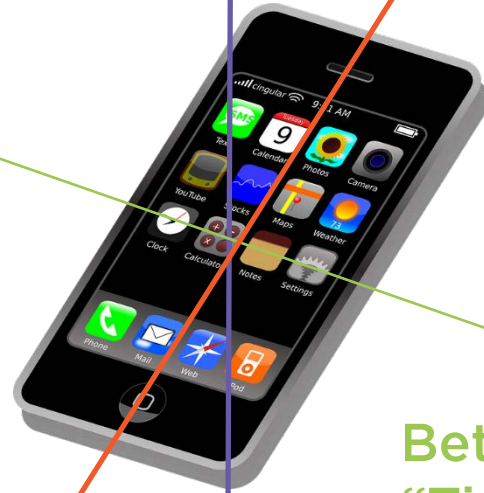
Gamma (Y)
“Tilting phone left/right”

X

Y

Z

Beta (X)
“Tipping phone towards/away”

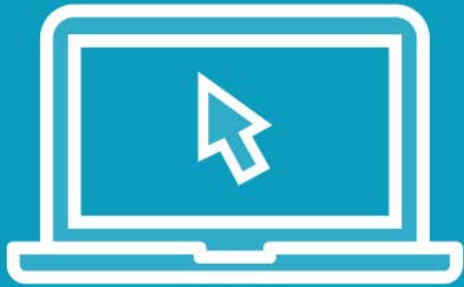


devicemotion

```
window.addEventListener  
("devicemotion", handle, true);  
  
function handle(e) {  
    e.acceleration; //meters per sec squared  
    e.accelerationIncludingGravity;  
    e.rotationRate; //degrees per sec  
    e.interval; interval in milliseconds  
}
```



Demo



Device Orientation Events



Device Orientation Event Demo

Device Orientation

absolute

false

alpha (z axis 0 to 360)

282.76081296032237

beta (x axis -180 to 180)

24.062696943337787

gamma (y axis -90 to 90)

-2.8287841693281934

Device Acceleration

acceleration

0.1619524508714676 -0.18358977138996124 -0.04928584024310112

accelerationIncludingGravity

0.5149292349815369 3.7745509147644043 8.940608024597168

rotationRate

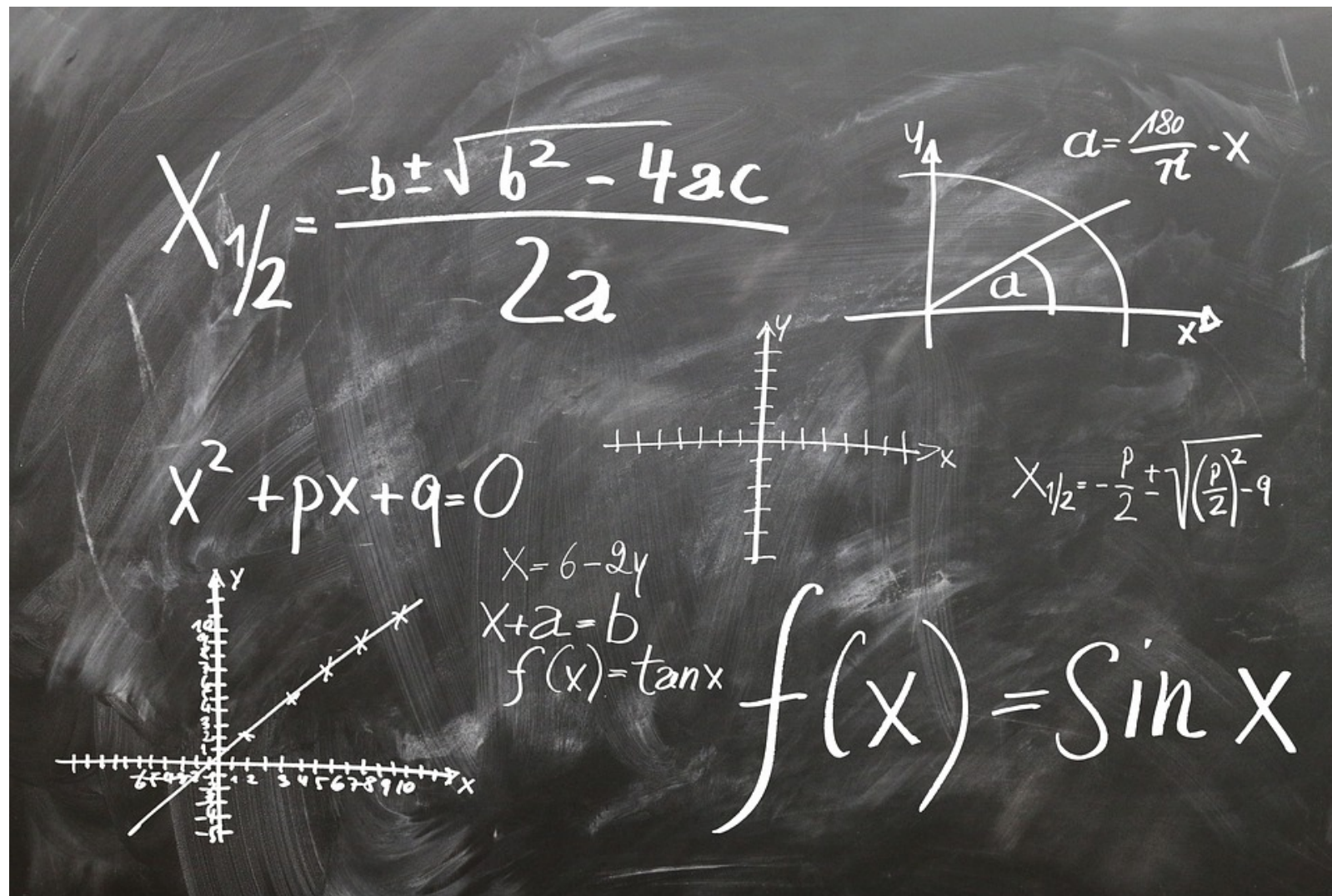
-0.000006214366294443607 0.011218131519854069 -0.03780003637075424

interval

16.666



Complexity Utilizing Data



Summary



Summary



WebVR

WebGL

Three.js

Device Orientation Events

