

WebVR

Declarative Virtual Reality in your browsers with a-frame

Carsten Sandtner (@casarock)

about:me

Carsten Sandtner
@casarock

Technical Director
mediaman //



Short History Of VR

View Master

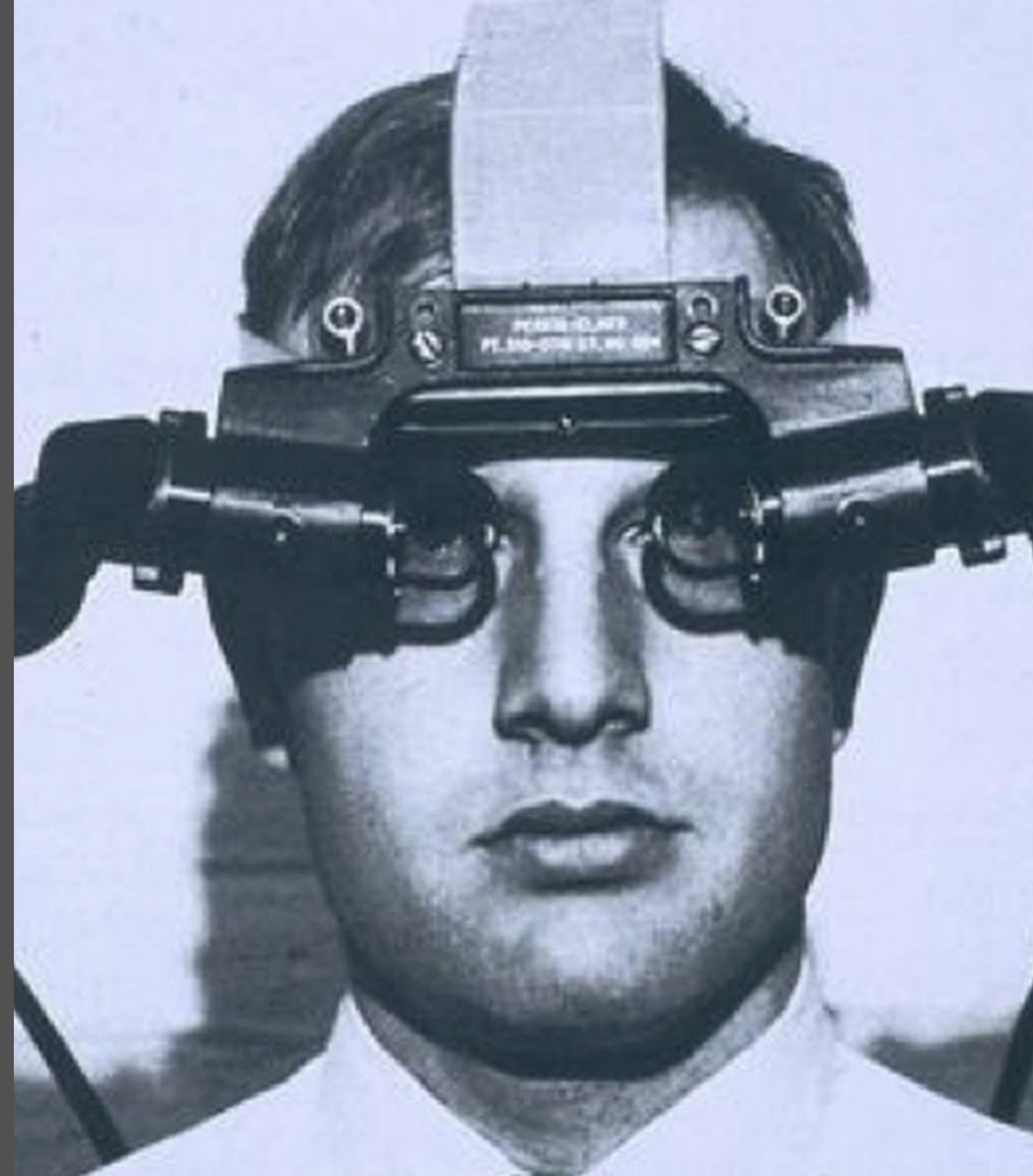
~1939



<https://flic.kr/p/hkLi6g>

Project Headsight

1961





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

Sensorama

~1962



Virtual Boy

1995



VR Today

Oculus Rift

DK1: 2013

DK2: 2014

Consumer Version: 2016



Google Cardboard

2014



HTC Vive

2016



Playstation VR

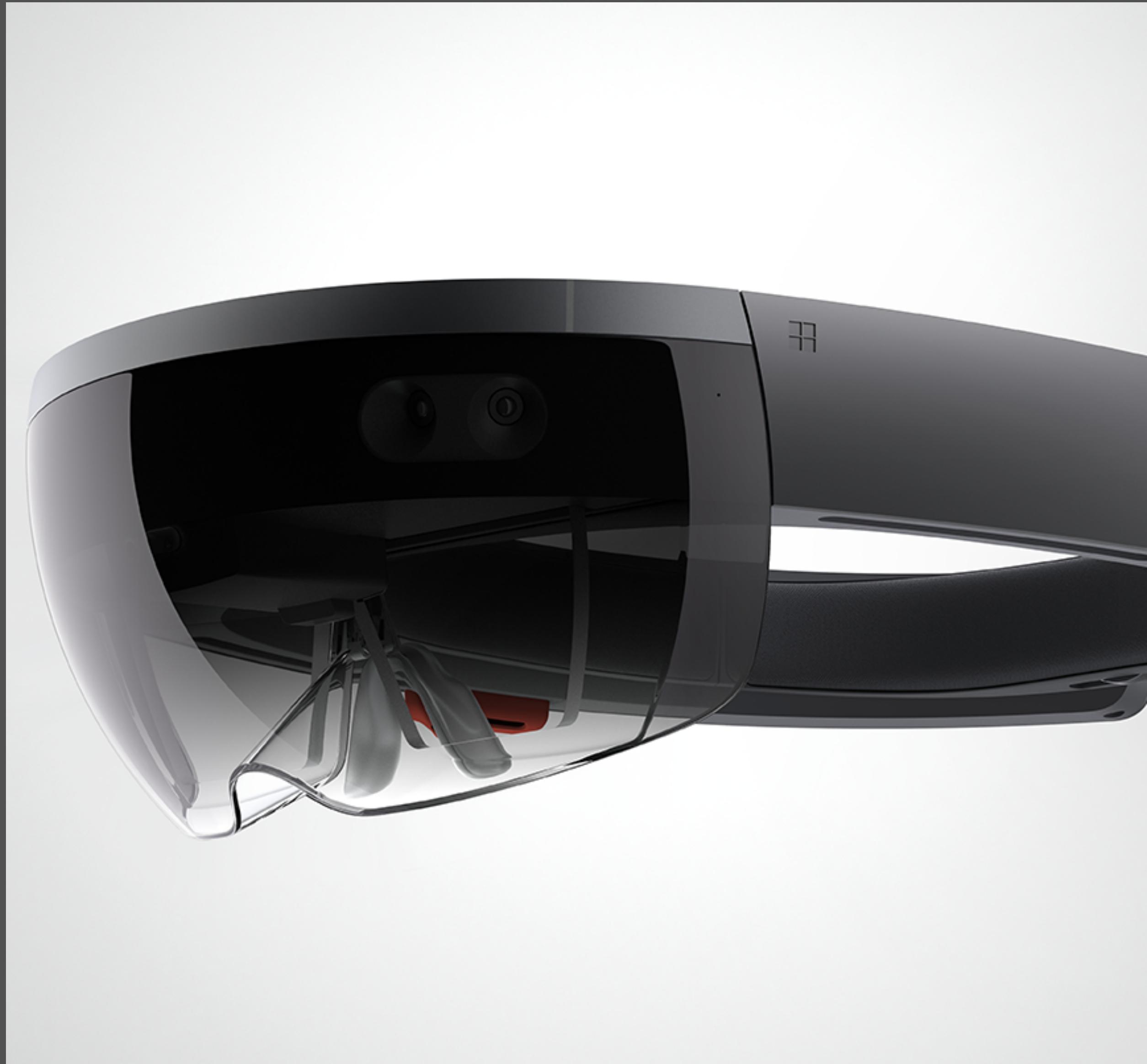
Oct. 2016



Microsoft Hololens

More AR, not really VR

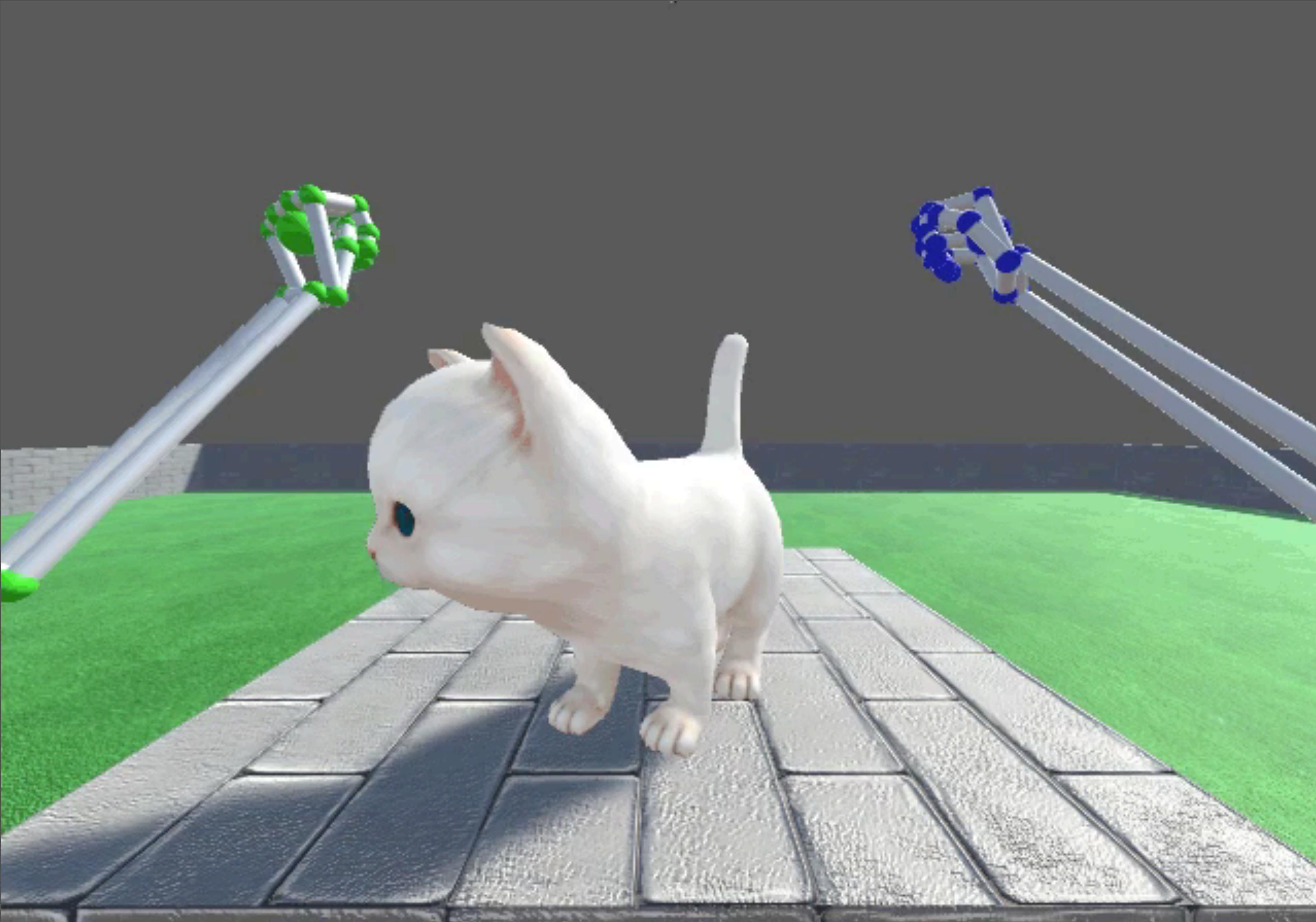
DevKit: 2016



VR In A Nutshell

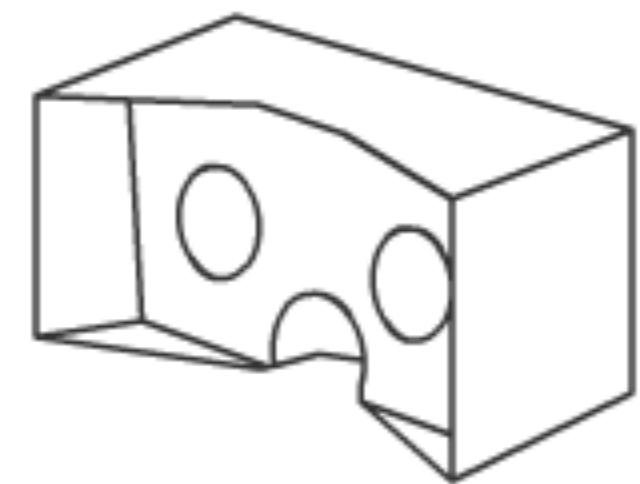


<https://twitter.com/guystuff/status/713075541738393600/video/1>



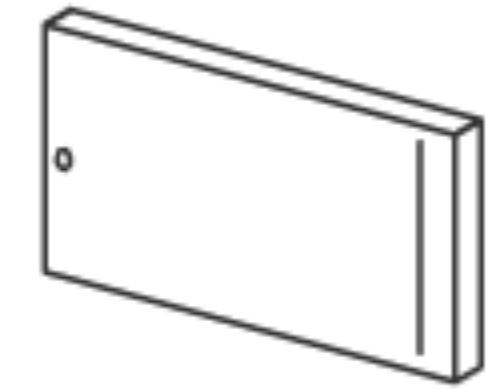
Mobile Based Setup

Mobile based VR setup



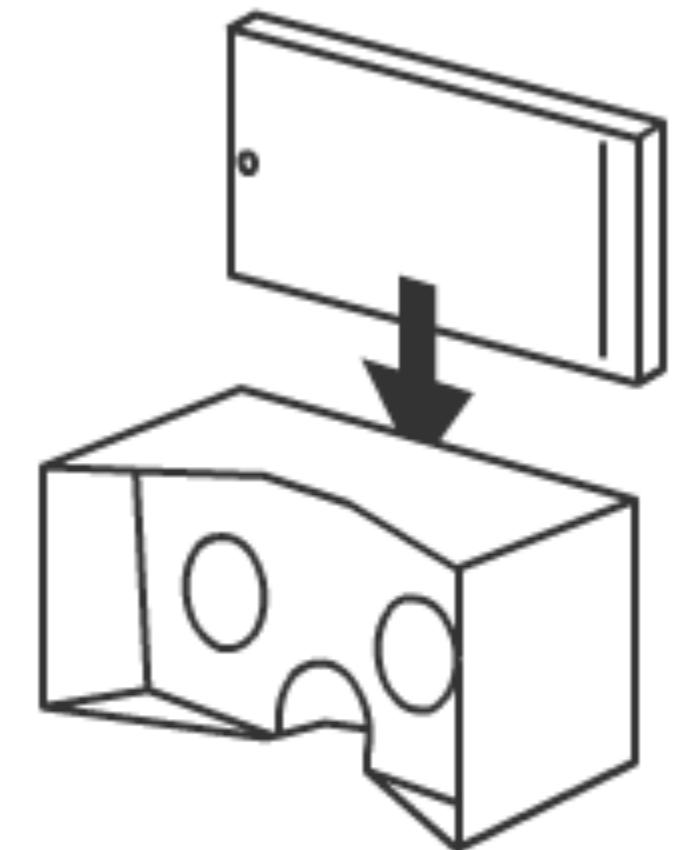
VR mount

+

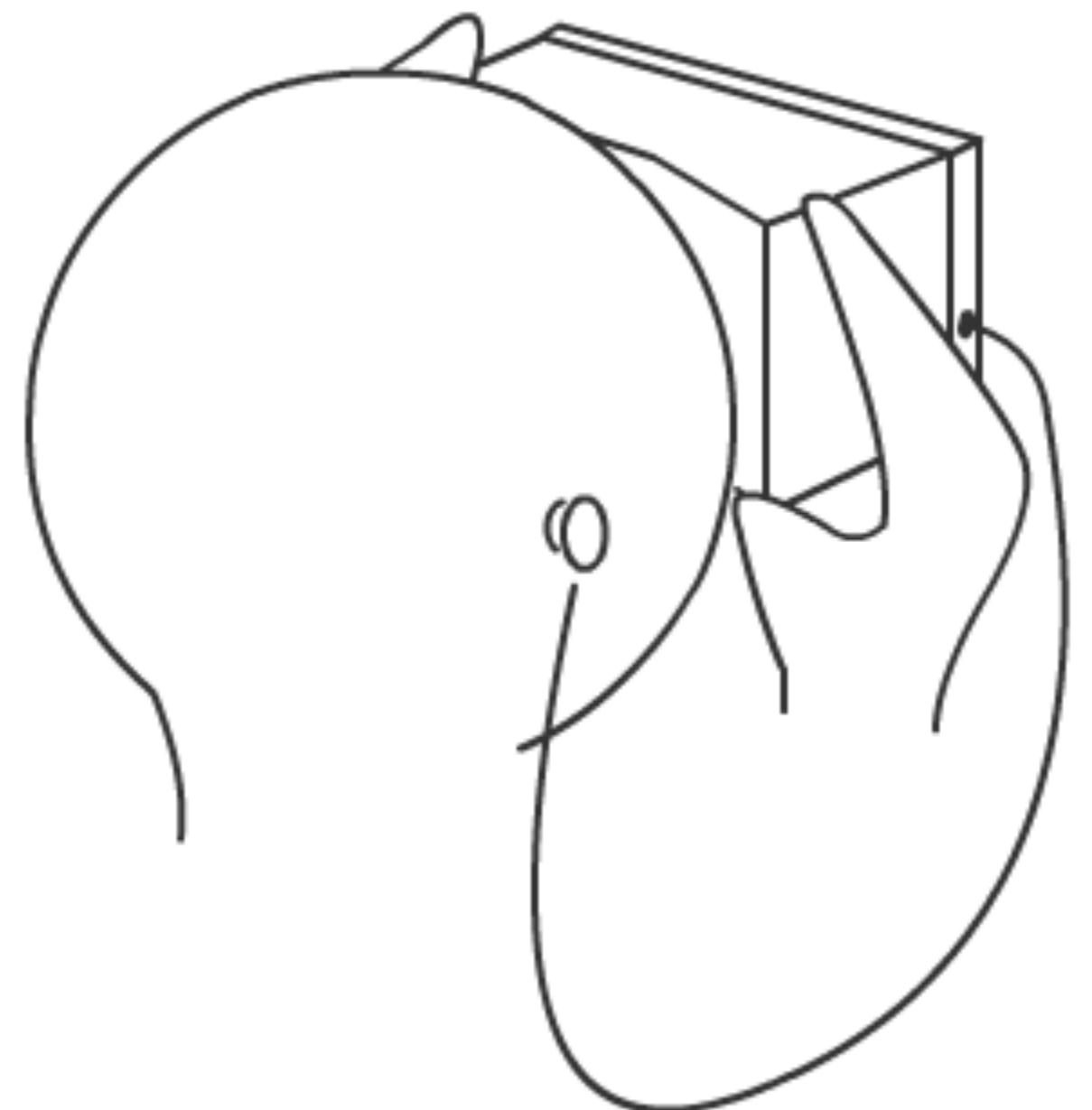


Smartphone

=

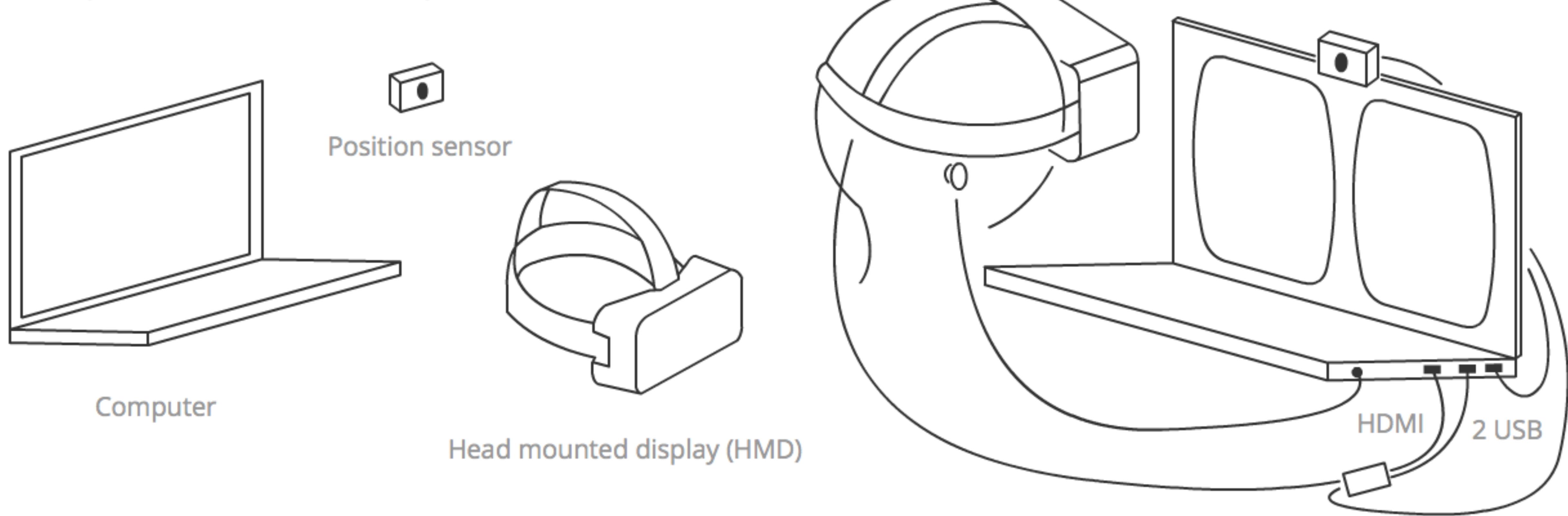


Head mounted display (HMD)



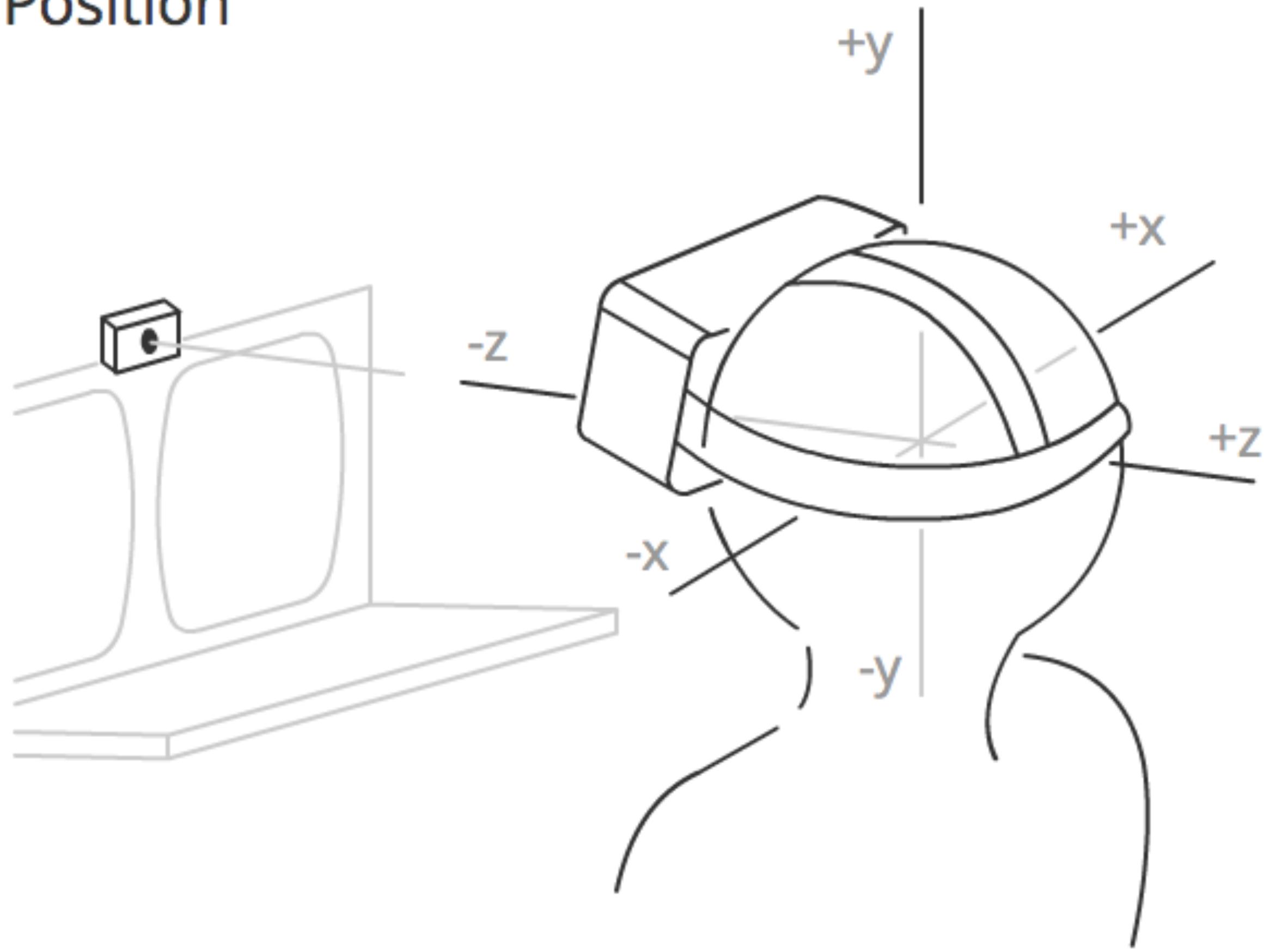
Computer Based Setup

Computer based VR setup

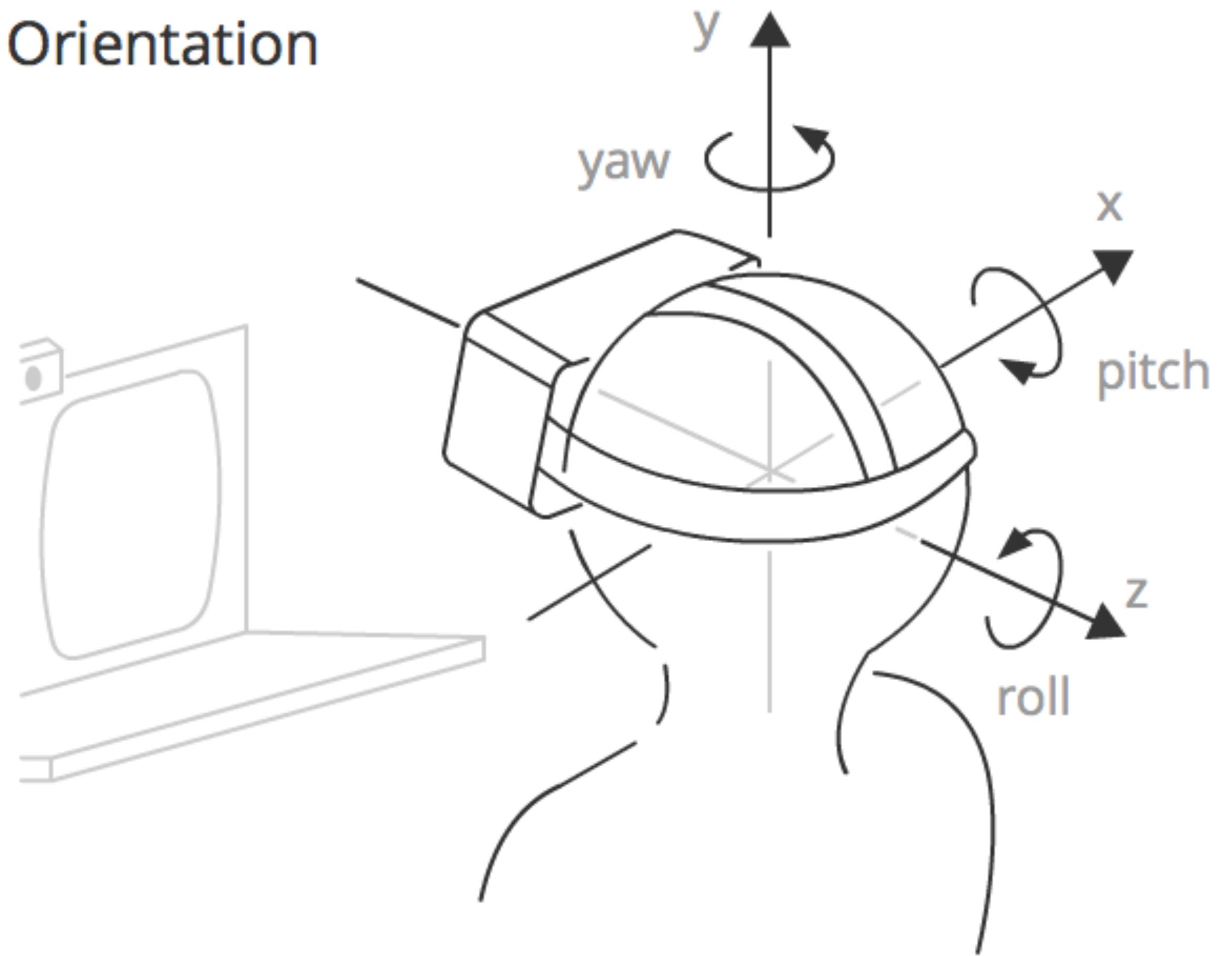


Sensors

Position

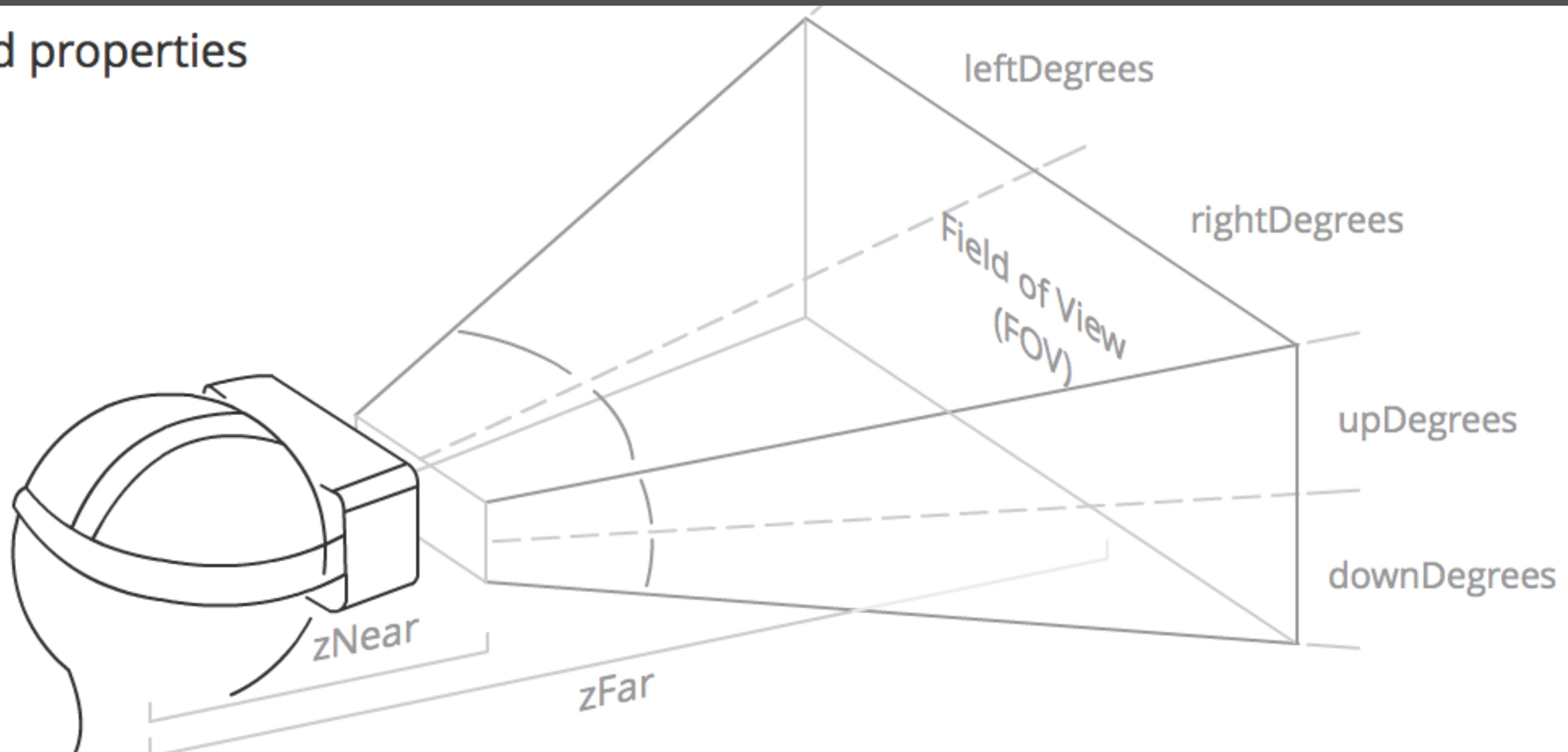


Orientation



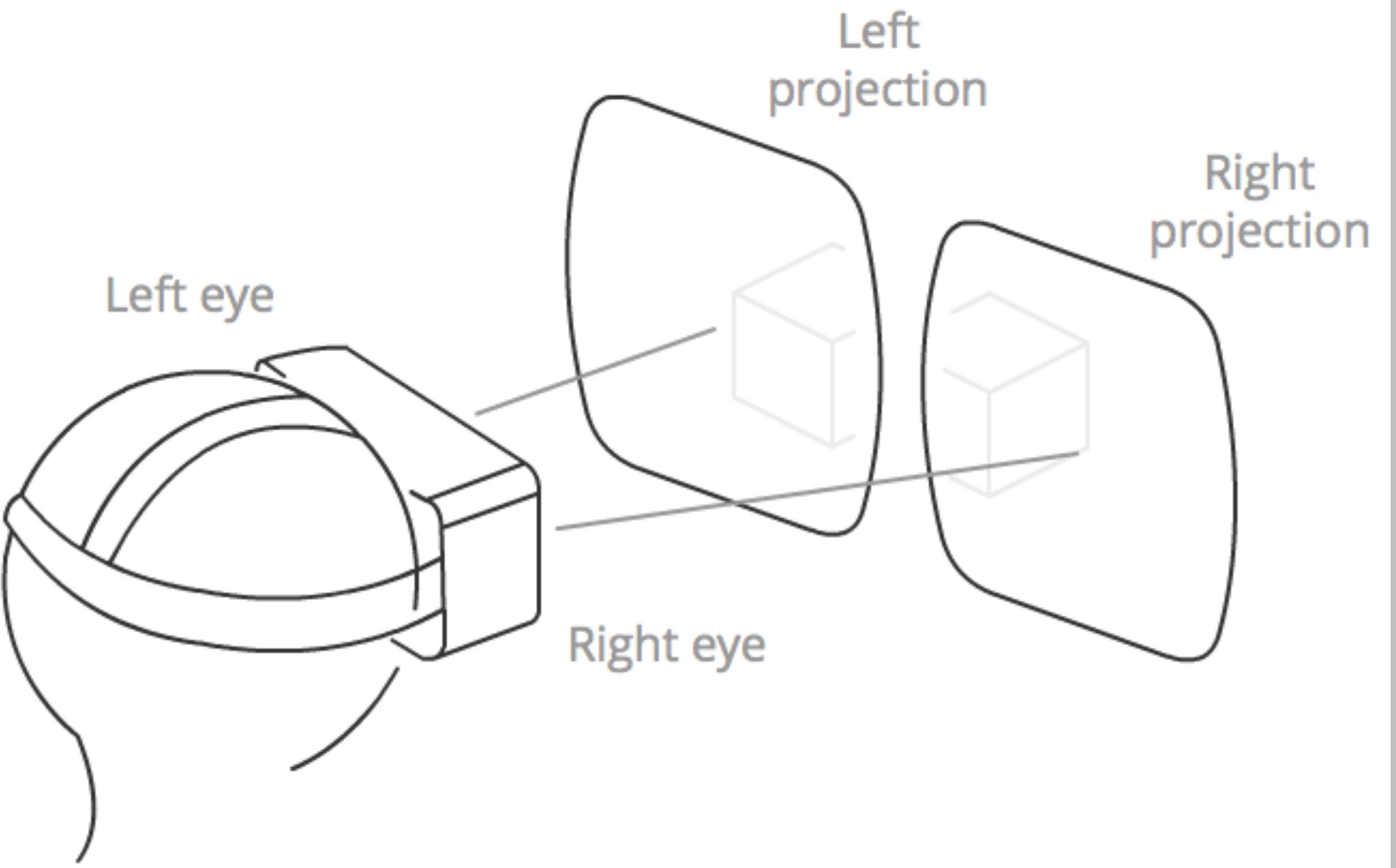
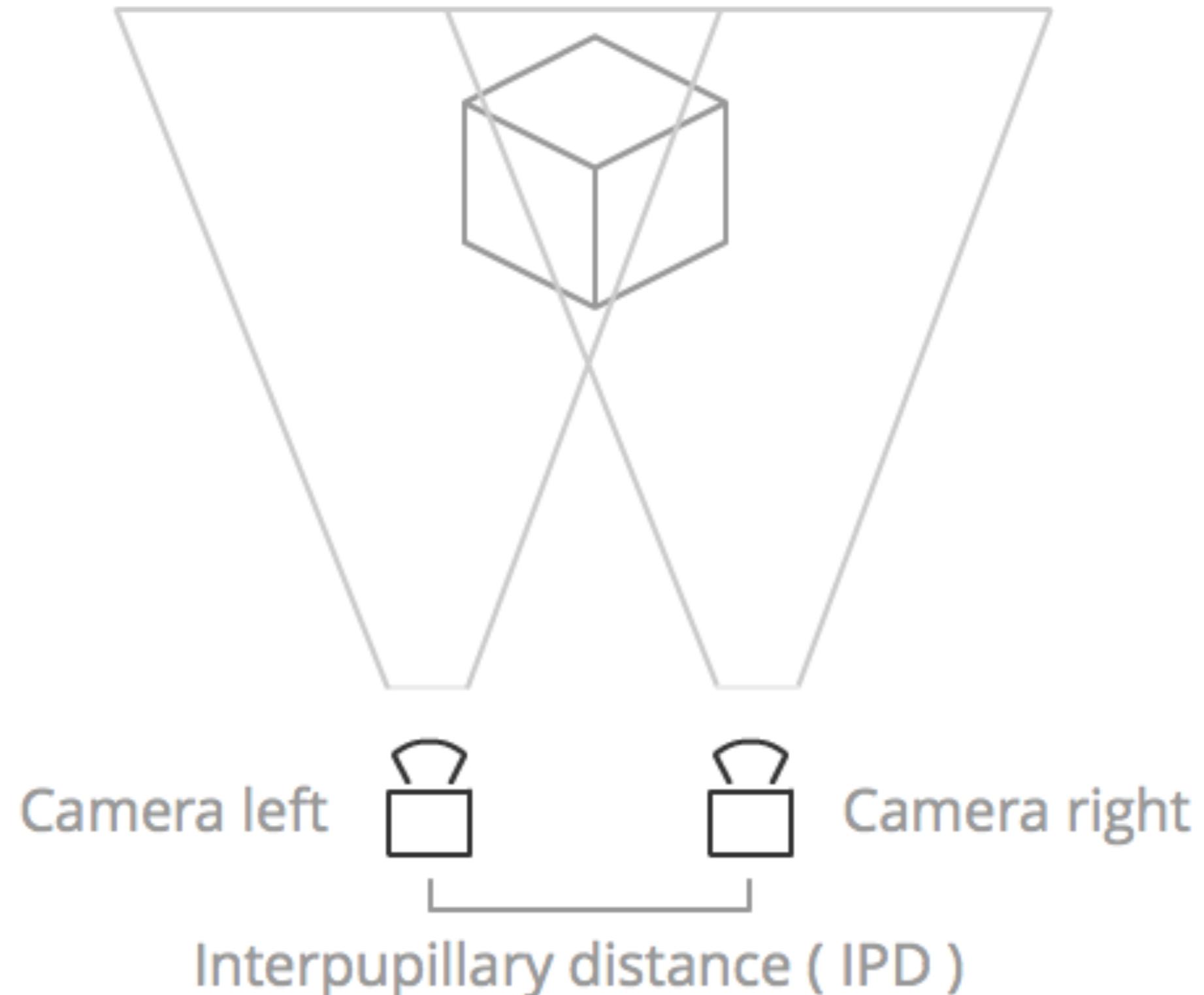
Field Of View

FOV related properties



Concepts For VR Apps

How to create stereoscopic 3D images



Concepts For VR Apps

- Eye strain
- Motion Sickness
- Latency
- FPS
- Degrees of Freedom (DoF)
- Cone of focus
- 3D Positional Audio -> **Web Audio API!**



<https://media.giphy.com/media/3o6gaVAXUrXIxEFYpWw/giphy.gif>

What Is WebVR?

Disclaimer

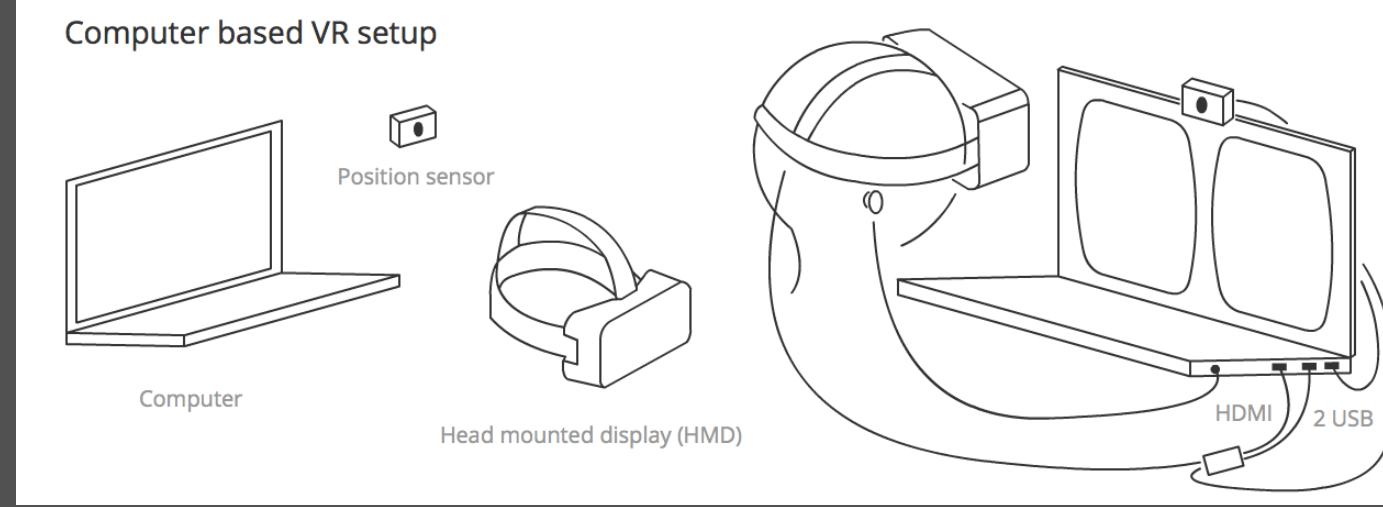
Currently an Editors Draft! (V1.0 since September, 1st 2016)

<https://mozvr.github.io/webvr-spec/>

Available APIs

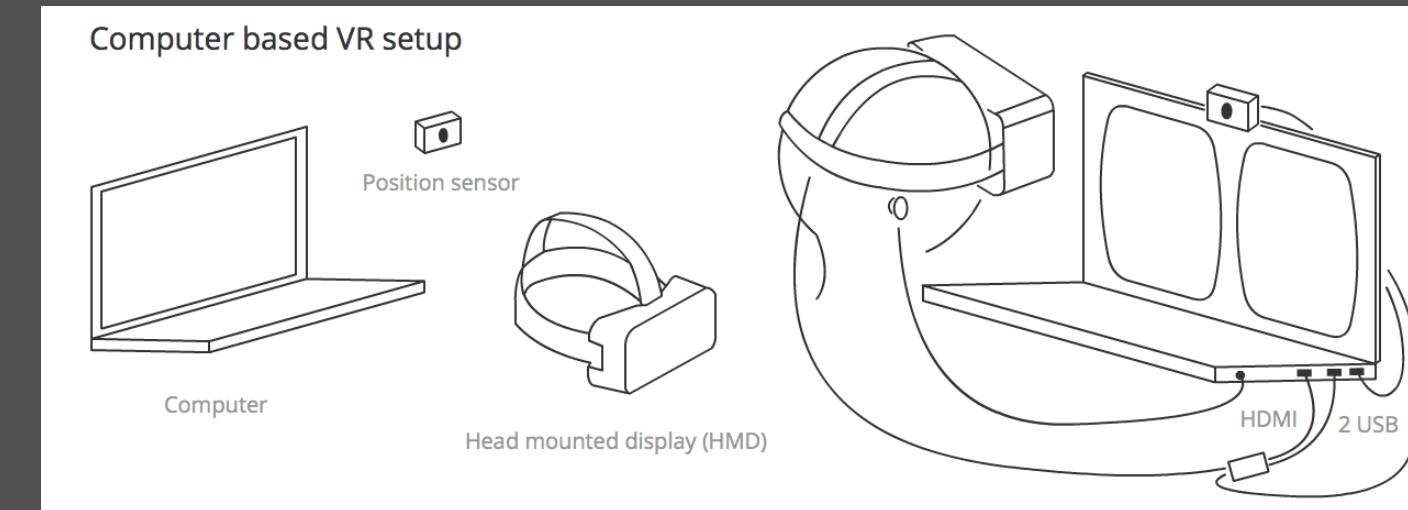
- Navigator.getVRDevices
- VRDevice/HMDVRDevice
- PositionSensorVRDevice
- VRPositionState
- VREyeParameters
- VRFieldOfView/VRFieldOfViewReadOnly

Get VR Devices



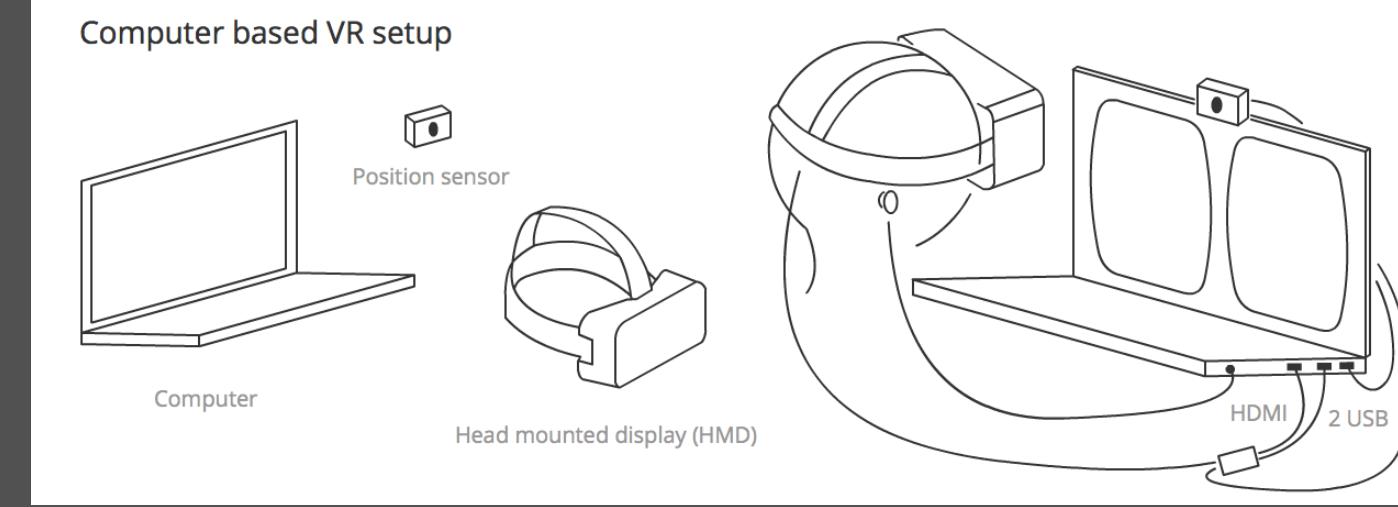
```
navigator.getVRDevices().then(function(devices) {  
  // Handle found Devices here...  
}) ;
```

(HMD) VR Device



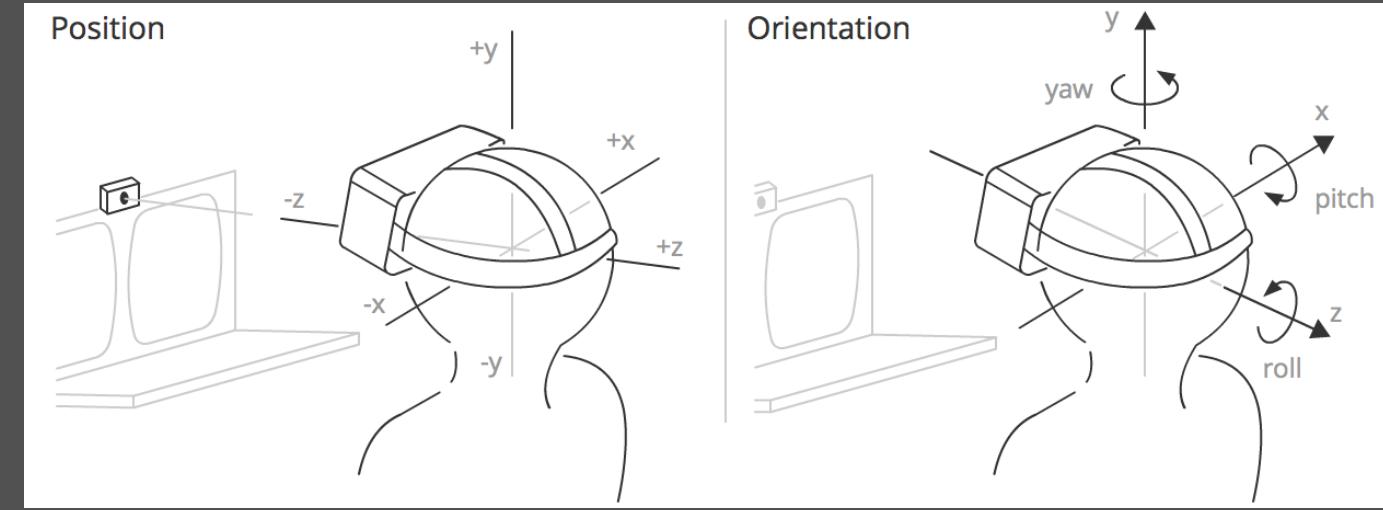
```
for (var i = 0; i < devices.length; ++i) {  
    if (devices[i] instanceof HMDVRDevice) {  
        gHMD = devices[i];  
        break;  
    }  
}
```

Position Sensor



```
// If device found, get Position Sensor.  
if (gHMD) {  
    for (var i = 0; i < devices.length; ++i) {  
        if (devices[i] instanceof PositionSensorVRDevice  
        && devices[i].hardwareUnitId === gHMD.hardwareUnitId)  
        {  
            gPositionSensor = devices[i];  
            break;  
        }  
    }  
}
```

Position State



```
var posState = gPositionSensor.getState();
if (posState.hasPosition) {
    posPara.textContent = 'Position: x' + (posState.position.x) +
        ' y' + (posState.position.y) +
        ' z' + (posState.position.z);
}

if (posState.hasOrientation) {
    orientPara.textContent = 'Orientation: x' + (posState.orientation.x) +
        ' y' + (posState.orientation.y) +
        ' z' + (posState.orientation.z);
}
```


Stereoscopic Rendering in WebGL

```
/*
https://hacks.mozilla.org/2015/09/stereoscopic-rendering-in-webvr/
*/
function update() {
    // ... other stuff happens here ...
    // left eye
    gl.viewport(0, 0, canvas.width / 2, canvas.height);
    mat4.multiply(mvpMatrix, leftEyeProjectionMatrix, leftEyeViewMatrix);
    gl.uniformMatrix4fv(uniforms.uMVPMatrixLocation, false, mvpMatrix);
    gl.drawElements(mode, count, type, offset);

    // right eye
    gl.viewport(canvas.width / 2, 0, canvas.width / 2, canvas.height);
    mat4.multiply(mvpMatrix, rightEyeProjectionMatrix, rightEyeViewMatrix);
    gl.uniformMatrix4fv(uniforms.uMVPMatrixLocation, false, mvpMatrix);
    gl.drawElements(gl.TRIANGLES, n, gl.UNSIGNED_SHORT, 0);

    requestAnimationFrame(update);
}
```


three.js WebVR Renderer

```
<script src="js/three.min.js"></script>
<script src="js/effects/VREffect.js"></script>
<script src="js/controls/VRControls.js"></script>

<script>
  ...
  var effect = new THREE.VREffect( renderer );
  ...

  effect.render( scene, camera );
</script>
```


WebVR Boilerplate

three.js + webVRControls

<https://github.com/borismus/webvr-boilerplate>

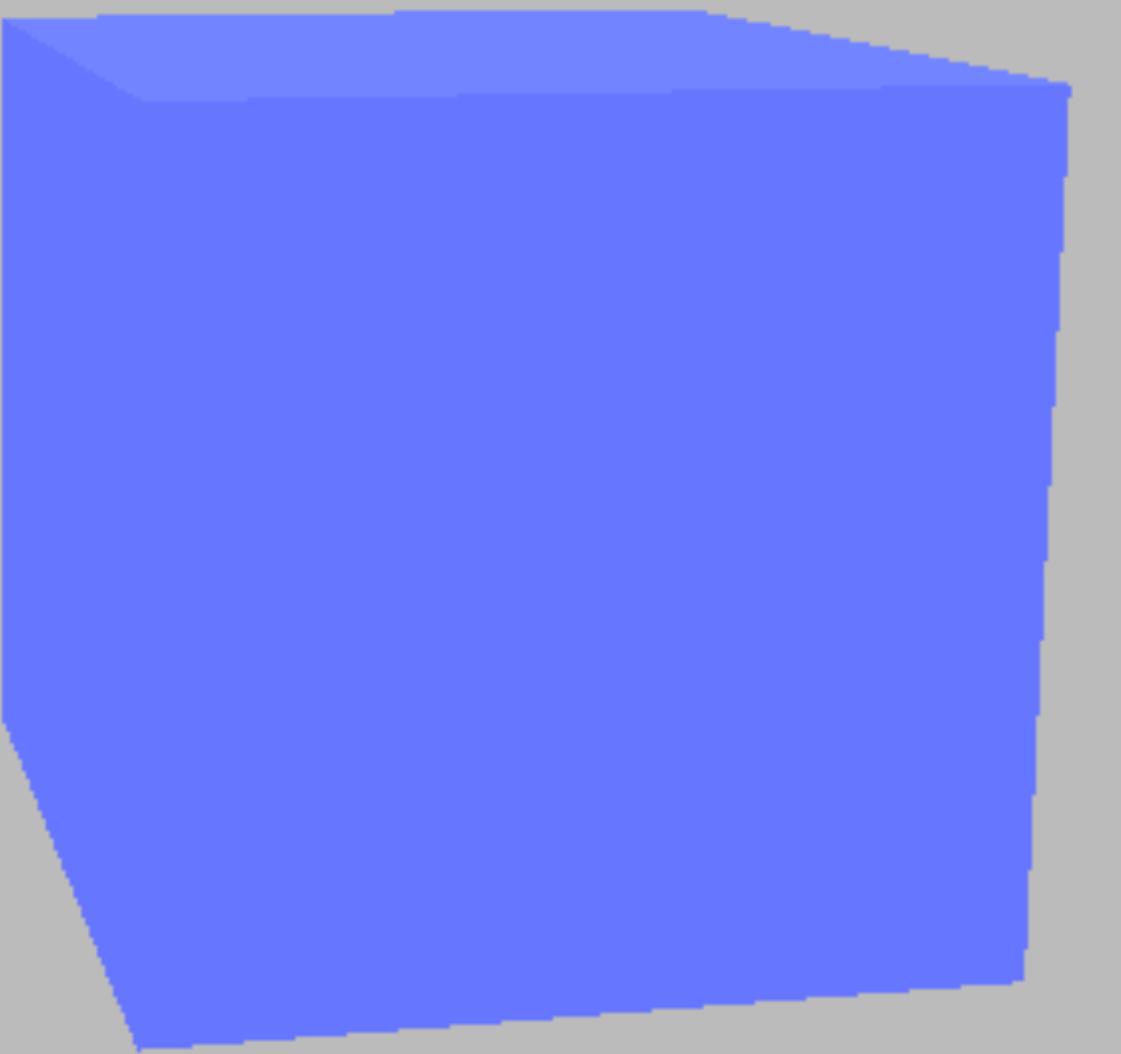
But: „declarative“?

Mozilla A-Frame

Building blocks for the virtual reality web

Hello World

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <title>Hello, World!</title>
    <script src="https://aframe.io/releases/0.2.0/aframe.min.js"></script>
  </head>
  <body>
    <a-scene>
      <a-box id="mybox" color="#6173F4" width="1" height="1"
             depth="1" position="1 1 1" rotation="0 0 0" scale="1 1 1">
      </a-box>
      <a-sky color="#bbb"></a-sky>
    </a-scene>
  </body>
</html>
```



Animated Box

```
<!DOCTYPE html>
<html>
  <head> . . . </head>
  <body>
    <a-scene>
      <a-box id="mybox" color="#6173F4" width="1" height="1"
             depth="1" position="1 1 1" rotation="0 0 0" scale="1 1 1">
        <a-animation attribute="rotation" repeat="indefinite" to="0 180 0"></a-animation>
      </a-box>
    </a-scene>
  </body>
</html>
```



Pointer

```
<!DOCTYPE html>
<html>
  <head> . . . </head>
  <body>
    <a-scene>
      <a-box id="mybox" color="#6173F4" width="1" height="1"
             depth="1" position="1 1 1" rotation="0 0 0" scale="1 1 1">
        <a-animation attribute="rotation" repeat="indefinite" to="0 180 0"></a-animation>
      </a-box>

      <a-camera position="0 0 0">
        <a-cursor color="#0000ff">
      </a-camera>
    </a-scene>
  </body>
</html>
```



o



Add Events

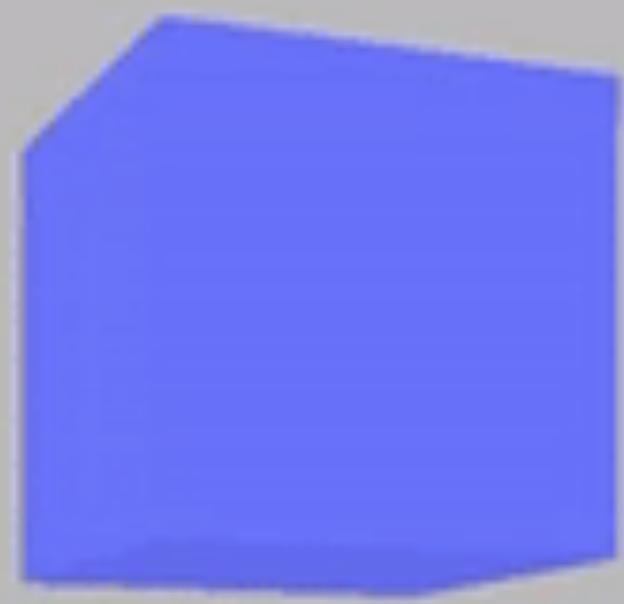
```
<!DOCTYPE html>
<html>
  <head> . . . </head>
  <body>
    <a-scene>
      <a-box id="mybox" color="#6173F4" width="1" height="1"
             depth="1" position="1 1 -5" rotation="0 0 0" scale="1 1 1">
        <a-animation attribute="rotation" repeat="indefinite" to="0 180 0"></a-animation>
        <a-event name="mouseenter" color="#ff0000"></a-event>
        <a-event name="mouseleave" color="#6173F4"></a-event>
      </a-box>
      <a-camera position="0 0 0">
        <a-cursor color="#0000ff">
      </a-camera>
      <a-sky color="#bbb"></a-sky>
    </a-scene>
  </body>
</html>
```

Add Events (Pure JS)

```
var box = document.querySelector('#mybox');

box.addEventListener('mouseenter', function () {
    box.setAttribute('material', {color: '#ff0000'});
} );

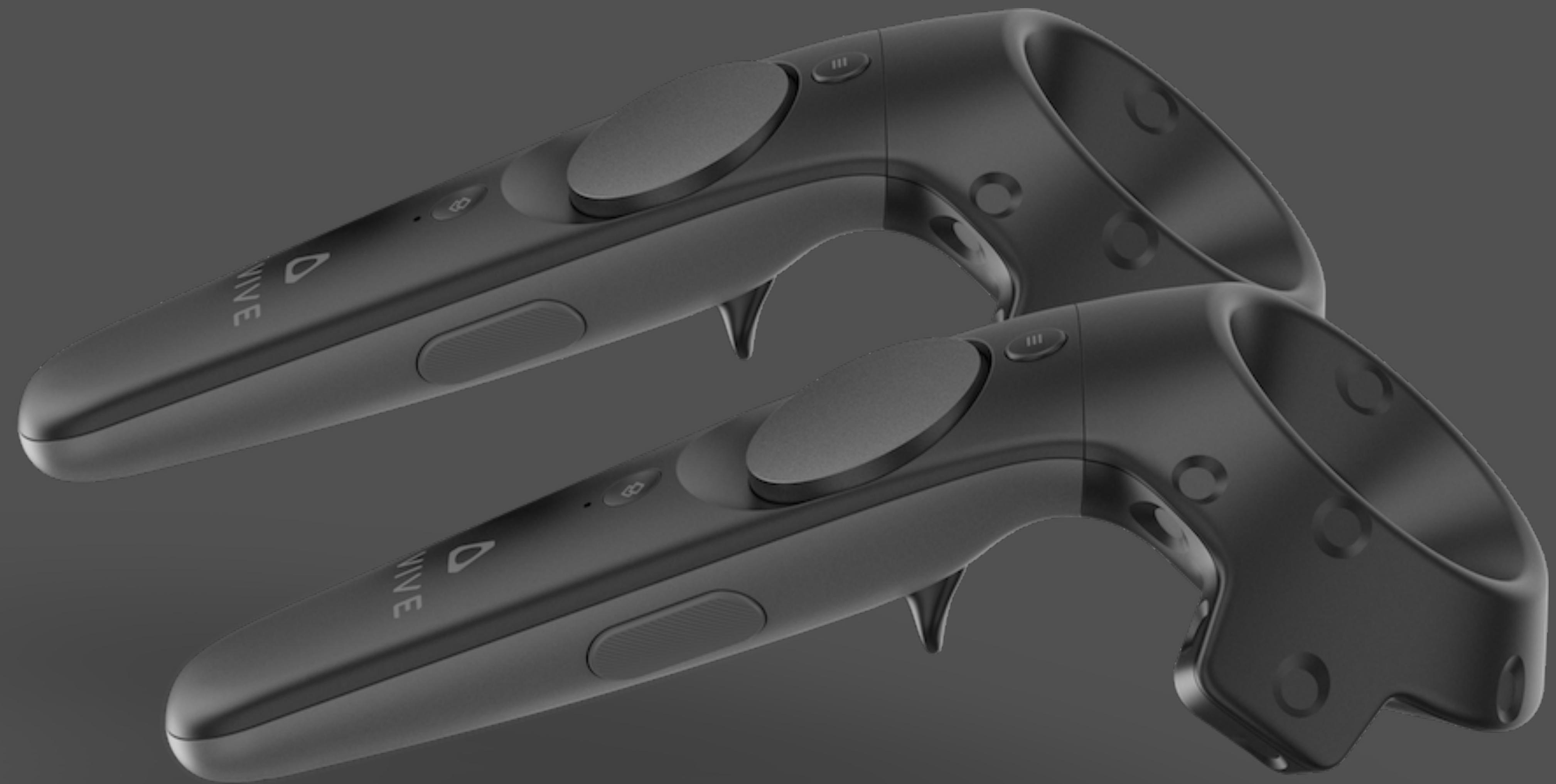
box.addEventListener('mouseleave', function () {
    box.setAttribute('material', {color: '#6173F4'});
} );
```

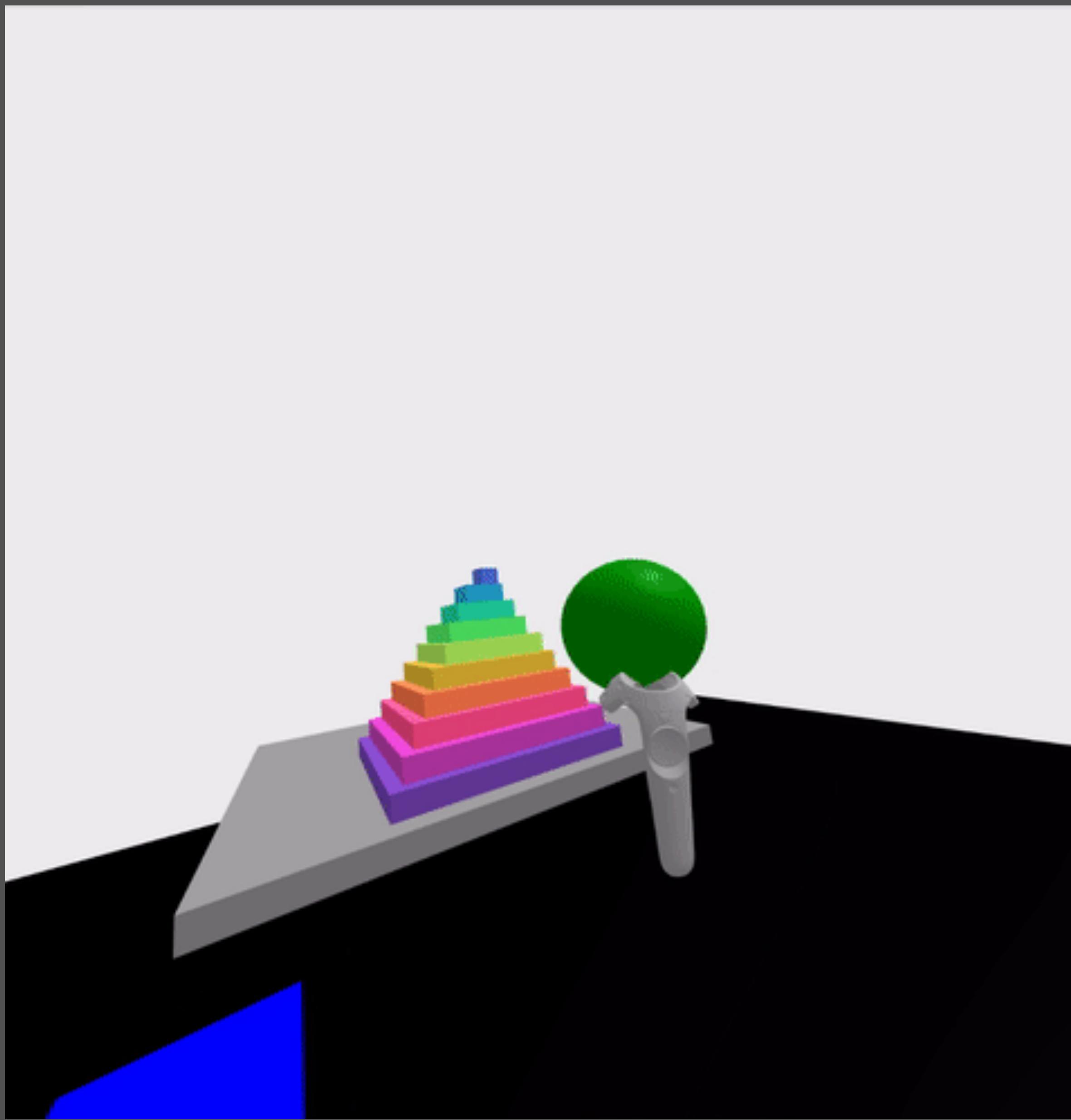


Input Devices?











Why WebVR?

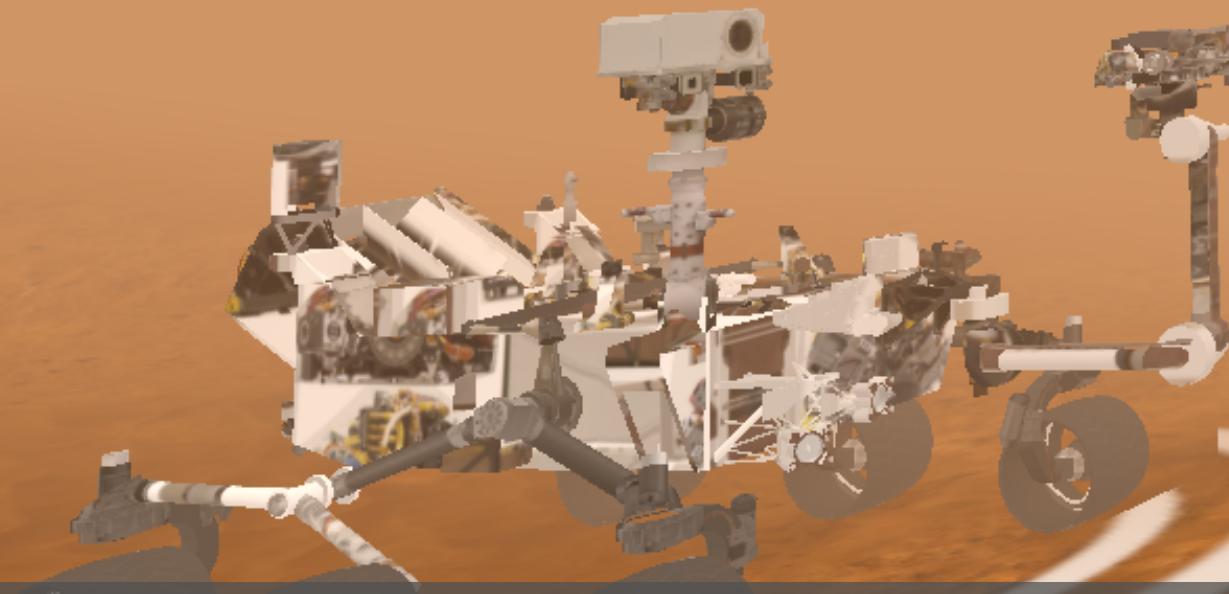
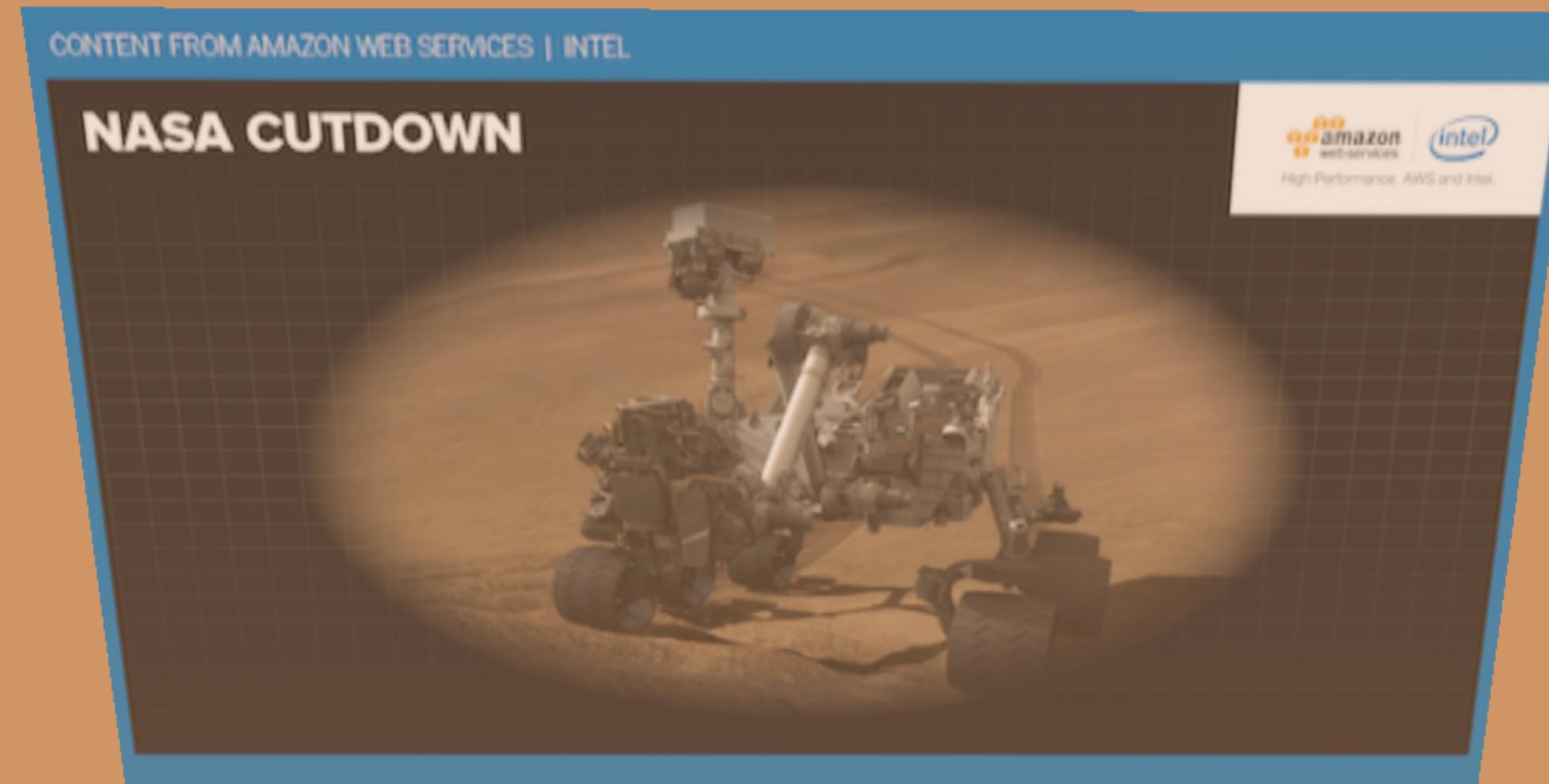
Some use cases

- Immersive reportages
- 3D Handbooks/Instructions
- Panorama view without ~~stupid~~ plugins
- Marketing instrument. (Supplements in magazines)

Some great examples

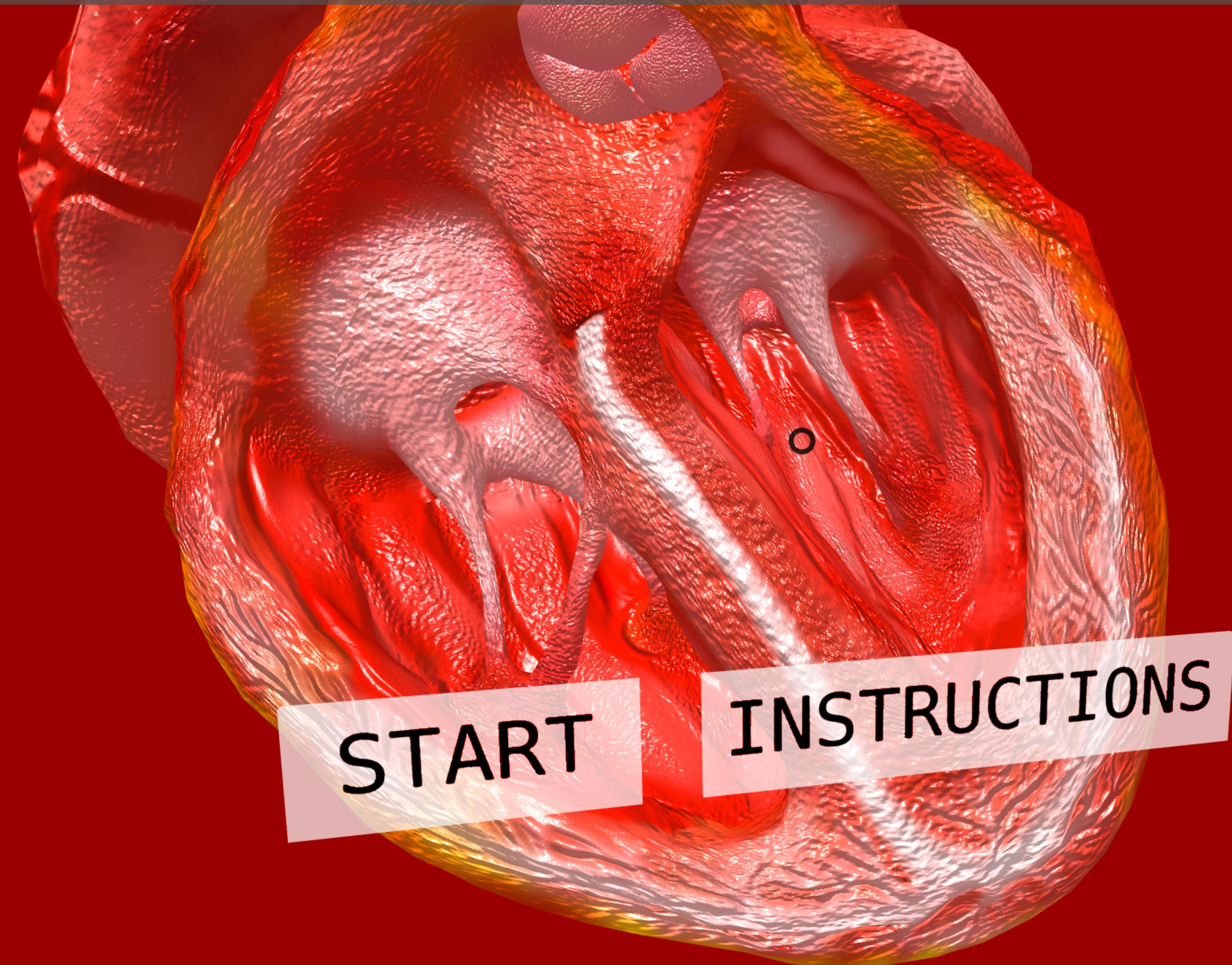
<http://www.360syria.com/>

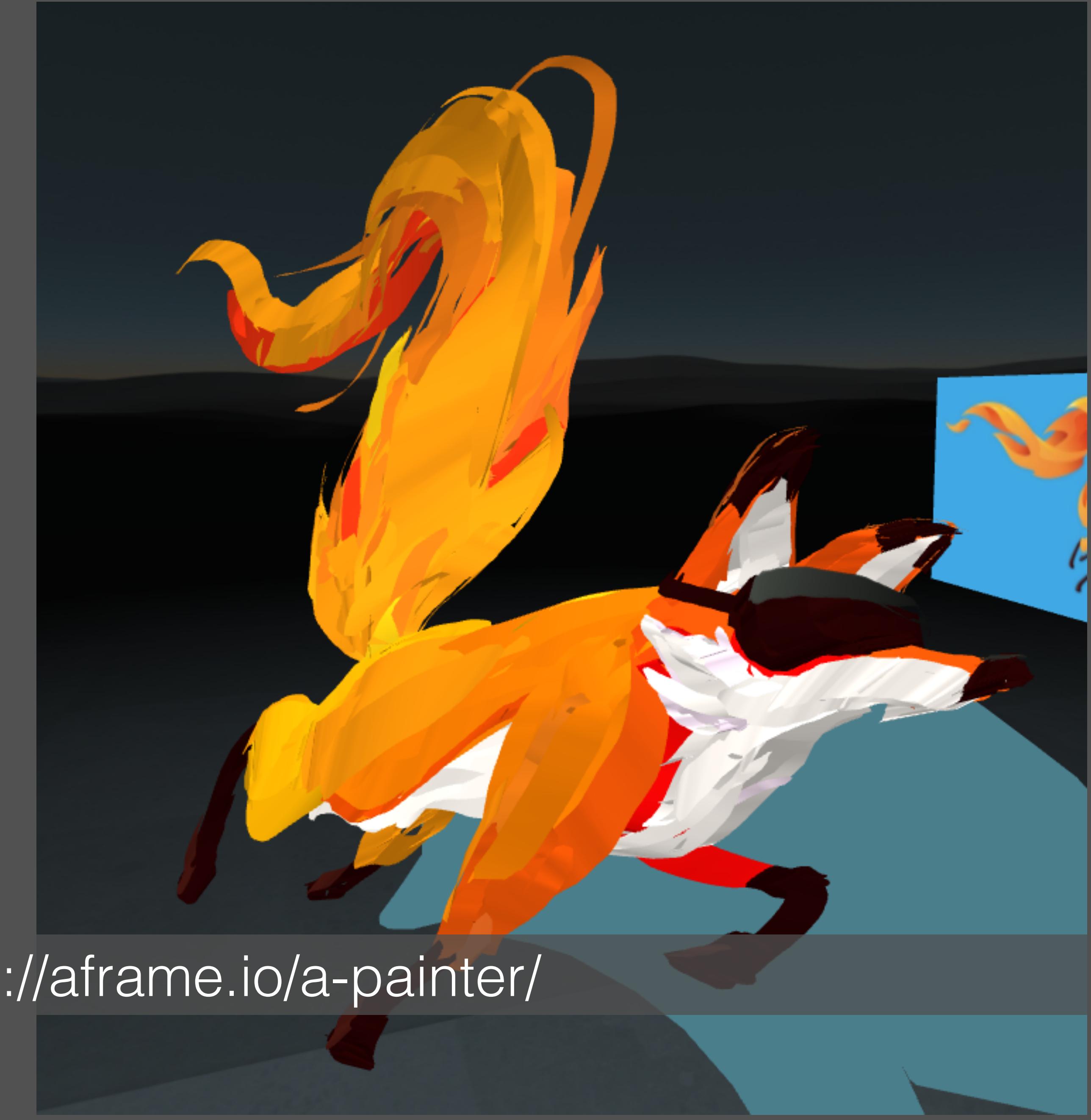




<https://www.washingtonpost.com/video/mars/public/>

https://s3.amazonaws.com/vr-asset-repo/heart_demo_slack.html





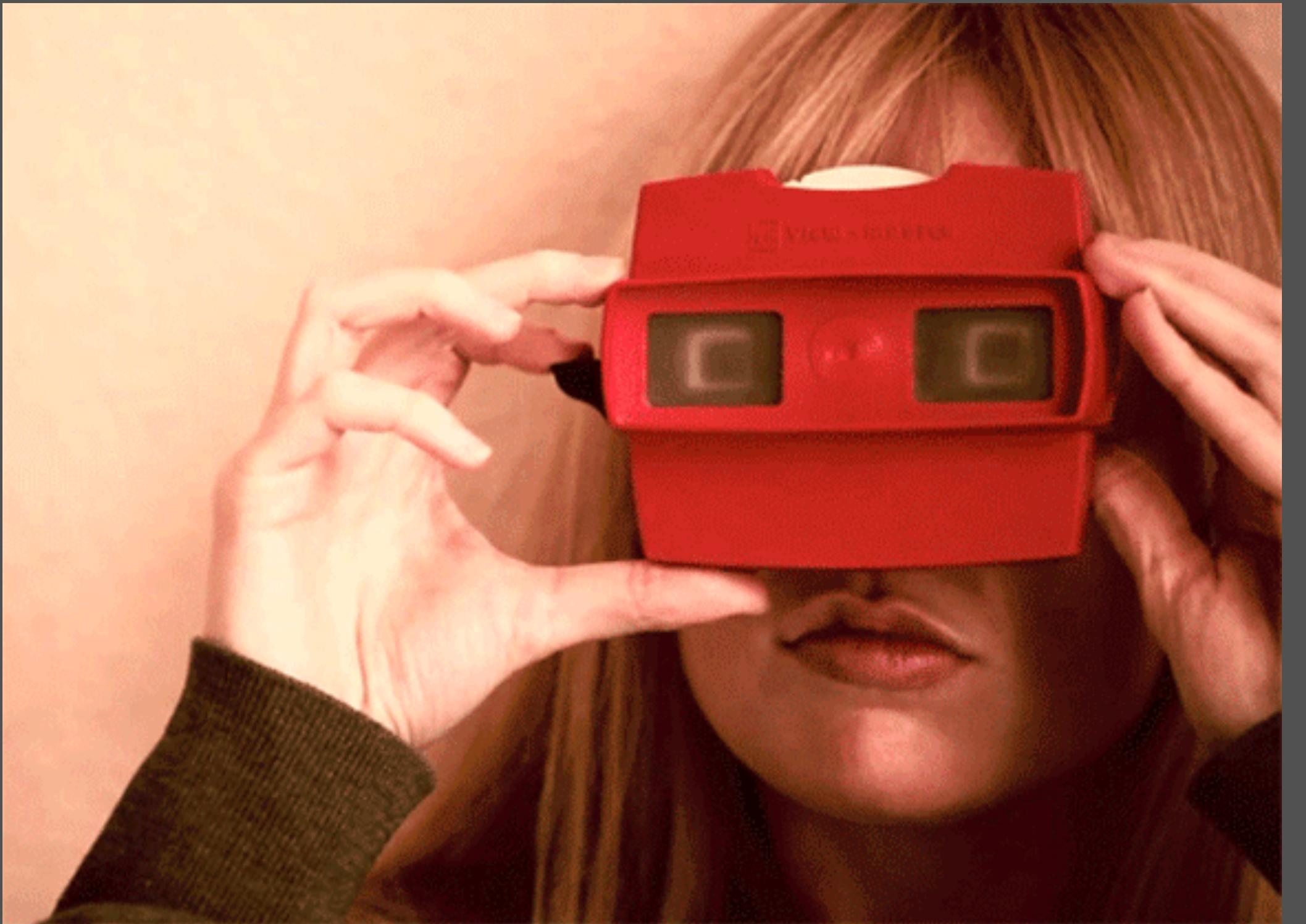
APainter - <https://aframe.io/a-painter/>

Resources

- <https://webvr.rocks/>
- <https://aframe.io/>
- [https://developer.mozilla.org/de/docs/Web/API/WebVR API](https://developer.mozilla.org/de/docs/Web/API/WebVR_API)

Conclusion

VR is amazing



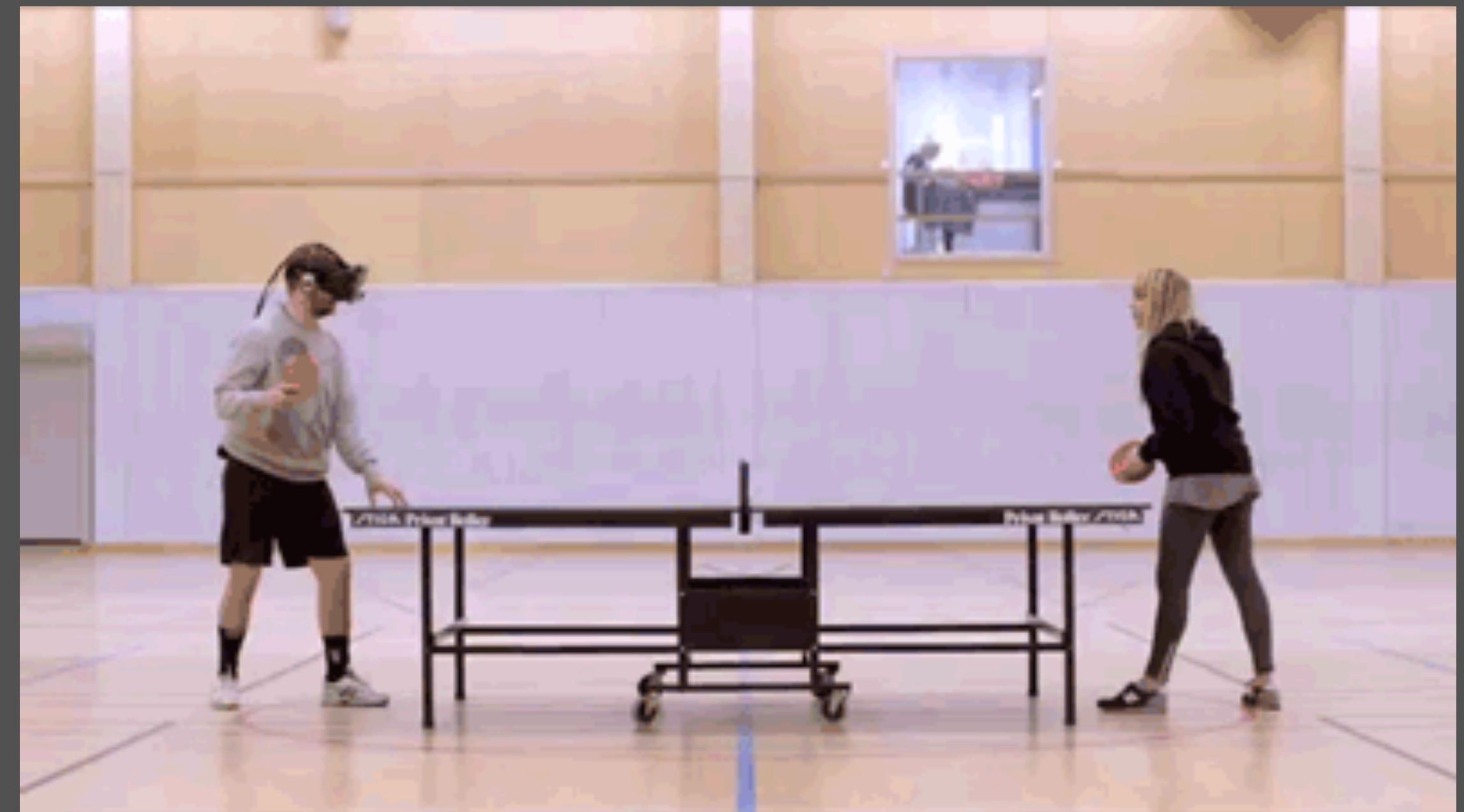
WebVR is amazing...

... but it's not ready
(Editors Draft, Browser support)

... and has high Hardware Requirements!

... HMD Devices are not cheap.
(Except: Google Cardboard)

... and it's a pleasure to create content!



WebVR *is* amazing

Carsten Sandtner

@casarock

sandtner@mediaman.de

