WebGL:
the next generation

Tony Parisi
VP, Web and Open Technologies
Wevr

March, 2015

about me

CONTACT

tparisi@gmail.com
skype: auradeluxe
http://twitter.com/auradeluxe
http://www.tonyparisi.com/
http://www.learningwebgl.com/

GET GLAM

http://www.glamjs.org/
https://github.com/tparisi/glam/

GET THE BOOKS!

Learning Virtual Reality
http://www.amazon.com/Learning-Virtual-Reality-Experiences-Applications/dp/1491922834
Programming 3D Applications with HTML and WebGL
http://www.amazon.com/Programming-Applications-HTML5-WebGL-Visualization/dp/1449362966
WebGL: Up and Running
http://www.amazon.com/dp/144932357X

CREDS

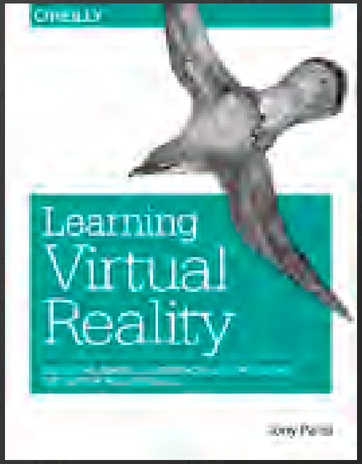
Co-creator, VRML and X3D

WORK

http://www.wevr.com/

MEETUPS

http://www.meetup.com/WebGL-Developers-Meetup/
http://www.meetup.com/Web-VR/



today's topics

- WebGL 2 – major upgrade to the standard
- WebVR – virtual reality in the browser, rendered with WebGL
- glTF – web-friendly 3D file format for use with WebGL

WebGL

the 3D rendering standard

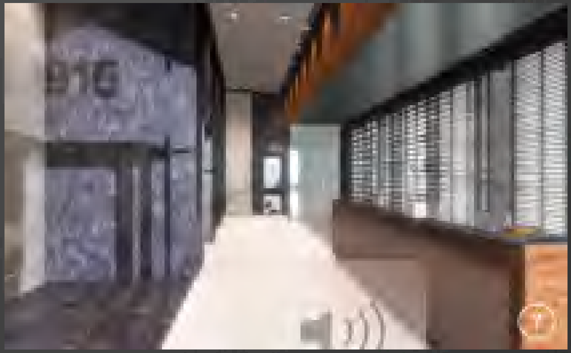
WebGL is on all desktop mobile browsers

3B seats.
Q.E.D.

WebGL

digital marketing

http://riskeverything.nike.com/



architecture

http://www.floored.com/blog/2015/webgl-real-time-physically-based-lighting

art

http://cabbi.bo/enough/

games

https://www.youtube.com/watch?v=io5snCcQ0ss

WebGL 2

major upgrade
based on OpenGL ES 3.0

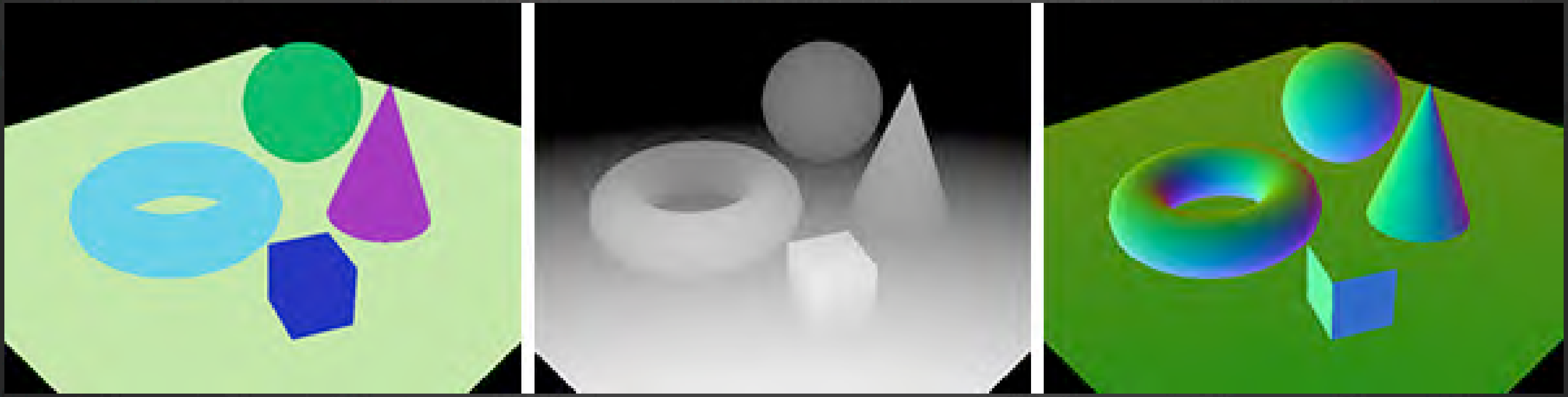
https://www.youtube.com/watch?v=2v6iLpY7j5M

WebGL 2

- promotes current WebGL extensions to full features
  - multiple render targets, geometry instancing, vertex array objects, fragment depth
- adds previously unsupported ES 3.0 features
  - multisampled render buffers
  - sampler objects
  - uniform buffers
  - 3D textures
  - profiling and debugging – sync objects, query objects
- some ES 3.0 features are **not** supported in WebGL 2
  - mapped buffers, program binaries, drawRangeElements()

deferred rendering example

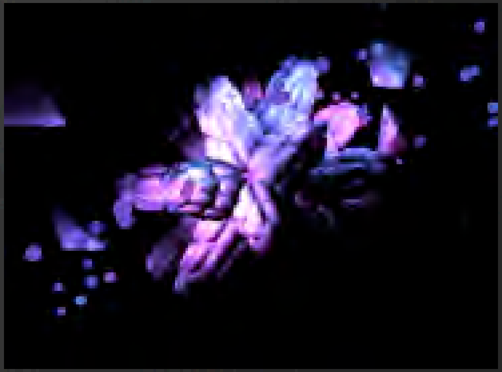
this technique is already being used in WebGL 1 with huge performance hit – three or more render targets. with multiple render targets you do the draw once instead of three or more times…



Color, Depth, and Normal buffers. (Images by astrofa, via Wikimedia Commons.)

Excellent example in WebGL1… would be even faster in V2!

http://marcinignac.com/blog/deferred-rendering-explained/demo/



WebGL 2

development status

- ○ enable WebGL 2 in Firefox

  https://wiki.mozilla.org/Platform/GFX/WebGL2

- ○ enable WebGL 2 in Chrome (Canary Windows/OSX, Dev Channel Linux)

  Run from command line with **--enable-unsafe-es3-apis**

- ○ specification

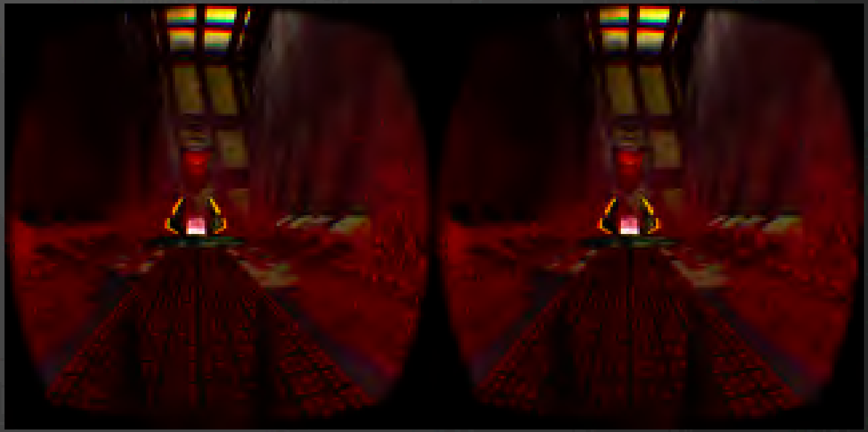
  https://www.khronos.org/registry/webgl/specs/latest/2.0/

- ○ live demo

  http://toji.github.io/webgl2-particles/

WebVR: virtual reality in the browser

- experimental WebVR API
  - Head-tracking and fullscreen VR support now in browser builds (nightly/dev channels)
  - awesome VR without big app downloads and installs!!!



quake 3 WebVR demo, developed by Brandon Jones of Google
http://media.tojicode.com/q3bsp/

http://mozvr.github.io/webvr-spec/

the WebVR API (1)

query for VR Display(s) for Rendering

enumerate available VR devices

```
var self = this;
var vrDisplay;
navigator.getVRDisplays().then( gotVRDisplays );
function gotVRDisplays ( displays ) {
  if (displays.length > 0) {
       vrDisplay = displays[0];

       self.left = vrDisplay.getEyeParameters( "left" );
        self.right = vrDisplay.getEyeParameters( "right" );
       self.vrDisplay = vrDisplay;
    }
}
```

get left/right eye (camera) information: horizontal offset,
field of view, viewport width;
we'll use WebGL to render the scene from two cameras

the WebVR API (2)

set up to present to the VR Display

VR presentation must be initiated by user action e.g. mouse click

```
someButton.addEventListener(’click', onStartPresent);
function onStartPresent () {
    vrDisplay.requestPresent({ source : webGLCanvas });
}
```

the WebGL canvas contains the rendered content to be presented on the VR display

the WebVR API (3)

render

WebVR introduces a new version of requestAnimationFrame() specifically for VR devices, making >60FPS rendering possible!

```
vrDisplay.requestAnimationFrame(runloop);
function runloop() {
    // set up for the next frame
    vrDisplay.requestAnimationFrame(runloop);

    // render the content
    var pose = vrDisplay.getPose();
    if (vrDisplay.isPresenting) {
            renderScene(pose, "left”);
            renderScene(pose, ”right"));
    }
    vrDisplay.submitFrame(pose);
}
```

get HMD position/ orientation

render scene once for each eye

submit rendered content to the HMD

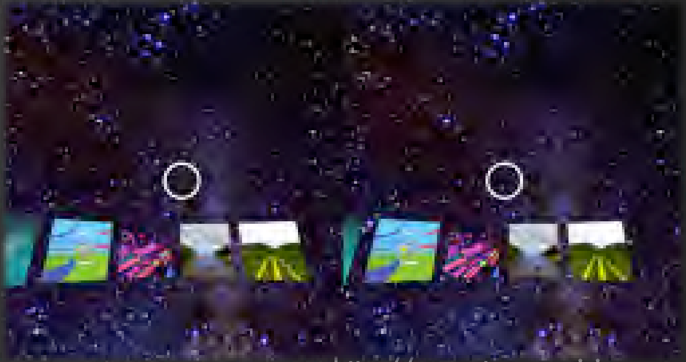
WebVR and mobile

- Google Cardboard Showcase
  - Mobile Chrome http://g.co/chromevr
- two ways to implement
  - for existing mobile browsers – render WebGL Side-by-Side stereo (no need to query devices), existing fullscreen and browser DeviceOrientation API
  - new WebVR API supported in betas of FF and Chrome

    http://mozvr.com/downloads/

    https://drive.google.com/folderview?id=0BzudLt22BqGRbW9WTHMtOWMzNjQ
- WebVR Polyfill – works across WebVR API and Cardboard styles

  https://github.com/borismus/webvr-polyfill



WebVR status and resources

- 1.0 specification – NOT a standard - still experimental

  http://mozvr.github.io/webvr-spec/

- Chromium builds

  https://drive.google.com/a/wevr.com/folderview?id=0BzudLt22BqGRbW9WTHMtOWMzNjQ&usp=sharing#list

- Brandon Jones' blog posting

  http://blog.tojicode.com/2016/02/moving-towards-webvr-10.html

- Casey Yee's introduction to WebVR

  https://hacks.mozilla.org/2016/03/introducing-the-webvr-1-0-api-proposal/

- mailing List

  web-vr-discuss@mozilla.org

- slack channel

  https://webvr.slack.com

gl Transmission **F**ormat
a “JPEG for 3D”

https://github.com/KhronosGroup/glTF

- runtime asset format for WebGL, OpenGL ES, and OpenGL applications
- compact representation for download efficiency
- loads quickly into memory
  - JSON for scene structure and other high-level constructs
  - GL native data types require no additional parsing
- full-featured
  - 3D constructs (hierarchy, cameras, shaders, animation, lights and standard materials via extensions)
  - full support for shaders and arbitrary materials
- runtime-neutral
  - can be created and used by any tool, app or runtime

the structure of a glTF file

```
"nodes": {
        "LOD3sp": {
            "children": [],
            "matrix": [
                    // matrix data here
                    ],
            "meshes": [
                "LOD3spShape-lib"
            ],
            "name": "LOD3sp"
        },
          …
"meshes": {
        "LOD3spShape-lib": {
            "name": "LOD3spShape",
            "primitives": [
                {
                    "attributes": {
                        "NORMAL": "accessor_25",
                        "POSITION": "accessor_23",
                        "TEXCOORD_0": "accessor_27"
                    },
                    "indices": "accessor_21",
                    "material": "blinn3-fx",
                    "primitive": 4
                }
            ]
        }
    },
```

scene structure defined as hierarchy of nodes

meshes and other visual types access low-level data

rich data e.g. vertices and animations stored in binary files

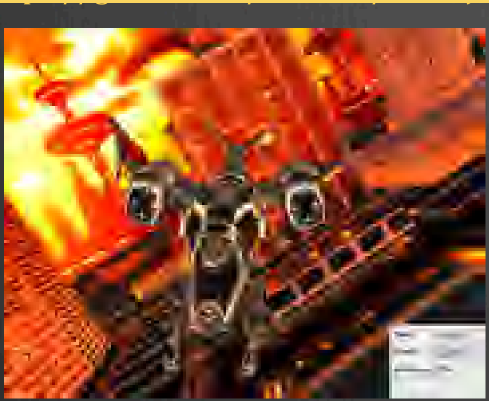
```
"buffers": {
       "duck": {
           "byteLength": 102040,
           "type": "arraybuffer",
           "uri": "duck.bin"
       }
   },
```

glTF™

adoption

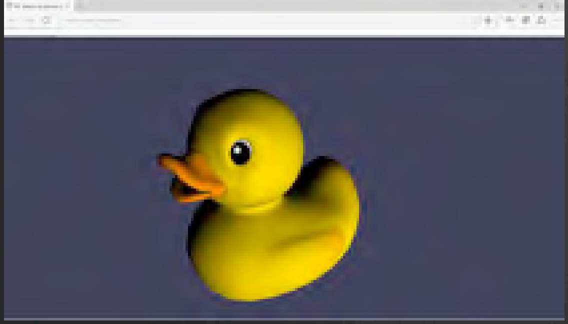
three.js Loader

https://github.com/mrdoob/three.js/



Babylon.js Loader (in development)

http://www.babylonjs.com/



Microsoft

CESIUM

It's the native format!

http://cesiumjs.org/

PIPELINE TOOLS

collada2gltf converter

https://github.com/KhronosGroup/glTF

Online drag and drop COLLADA to glTF converter

http://cesiumjs.org/convertmodel.html

FBX to glTF Converter

Drag and drop convertor coming

http://gltf.autodesk.io/

AUTODESK

WebGL ecosystem

game engines/IDEs

- Goo Engine * http://www.gootechnologies.com/
- Verold http://verold.com/ *
- Turbulenz https://turbulenz.com/
- PlayCanvas http://www.playcanvas.com/
- Sketchfab https://sketchfab.com/
- **Unreal * https://www.unrealengine.com/**
- **Unity * http://unity3d.com/#unity-5**

*** not open source**

scene graph libraries/page frameworks

- Three.js http://threejs.org/
- SceneJS http://scenejs.org/
- BabylonJS http://www.babylonjs.com/
- GLAM https://github.com/tparisi/glam
- A-Frame https://aframe.io/

WebGL: the next generation

Tony Parisi
VP, Web and Open Technologies
Wevr

March, 2015