WebGL

Patrick Cozzi
University of Pennsylvania
CIS 565 - Fall 2012

WebGL

- The web has text, images, and video
  - What is the next media-type?
- We want to support
  - Windows, Linux, Mac
  - Desktop and mobile

Bring 3D to the Masses

- Put it in on a webpage
  - Does not require a plugin or install
  - Does not require administrator rights
- Make it run on most GPUs

WebGL

- OpenGL ES 2.0 for JavaScript
  - Seriously, JavaScript

Image from http://www.khronos.org/assets/uploads/developers/library/2011-siggraph-mobile/Khronos-and-the-Mobile-

WebGL

- Includes
  - Vertex shaders
  - Fragment shaders
  - Vertex buffers
  - Textures
  - Framebuffers
  - Render states
  - …
- Does not include
  - Geometry shaders
  - Tessellation shaders
  - Vertex Array Objects
  - Multiple render targets
  - Floating-point textures
  - Compressed textures
  - FS depth writes
  - …

See http://www.khronos.org/registry/webgl/specs/latest/

WebGL

- If you know *OpenGL*, you already know *WebGL*
- If you know *C++*, the real learning curve is *JavaScript*

WebGL Alternatives?

- Flash
- Silverlight
- Java Applets
- Unity

WebGL

- Creating a context is easy:

```
// HTML:
<canvas id="glCanvas" width="1024"
  height="768"></canvas>

// JavaScript:
var gl =
  document.getElementById("glCanvas")
  .getContext("experimental-webgl");
```

WebGL

- The rest is similar to desktop OpenGL:

```
// ...
gl.bindBuffer(/* ... */);
gl.vertexAttribPointer(/* ... */);
gl.useProgram(/* ... */);
gl.drawArrays(/* ... */);
```

Checkout http://learningwebgl.com/

WebGL

- Create an animation loop:

```
(function tick(){
  // ... GL calls to draw scene
  window.requestAnimationFrame(tick);
})();
```

You want this to work cross-browser. See http://paulirish.com/2011/requestanimationframe-for-smart-animating/

WebGL Performance

- Performance can be very good. Why?

WebGL Performance

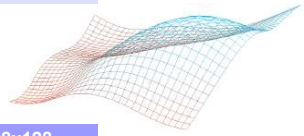
- Performance can be very good. Why?
  - The GPU is still doing the rendering
  - Batch!
    - Draw multiple objects with one draw call
    - Sort by texture
    - Push work into shaders
    - Push work into web workers

See http://www.youtube.com/watch?v=rfQ8rKGTVlg

WebGL Performance

	32x32	64x64	128x128
C++	1.9 ms	6.25 ms	58.82 ms
Chrome 18	27.77 ms	111.11 ms	454.54 ms
x slowdown	14.62	17.78	7.73

CPU-intensive



	32x32	64x64	128x128
C++	3.33 ms	9.43 ms	37.03 ms
Chrome 18	12.82 ms	22.72 ms	41.66 ms
x slowdown	3.85	2.41	1.13

GPU-intensive (256 draws per frame)

13

WebGL and other APIs

- Take advantage of other web APIs:
  - HTML5 <video>
  - 2D <canvas>
  - CSS transforms
  - Composite UI elements
  - Web workers
  - Typed Arrays

14

HTML5 on Mobile

- Touch events
  - Test with http://www.snappymaria.com/misc/TouchEventTest_v2.html
- Geolocation
- Device orientation and motion

- The future of HTML5 and WebGL on mobile is *very promising*

15

WebGL support is good, and it is getting better…

16

WebGL Stats

- In October, 2012

http://webglstats.com/

Desktop WebGL Support

- In October, 2012

- 3rd Party Plugins available

Android WebGL Support

- In October, 2012

Stock Browser
- Demo at SIGGRAPH 2011

For Cesium, see our mobile page

Mobile WebGL Support

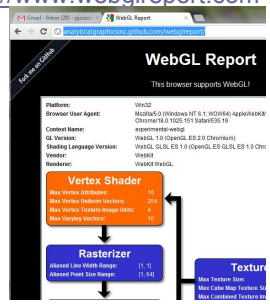
- In October, 2012

In iOS 5 for iAd developers

Windows Mobile

WebGL on Your System

- http://www.webglreport.com



Desktop WebGL Support

- Windows
  - No OpenGL driver installed? Old driver?
    - Only 35% of Windows XP machines have GL 2 drivers
  - Buggy driver?
  - No problem:
- *ANGLE* – *A*lmost *N*ative *G*raphics *L*ayer *E*ngine

OpenGL ES 2.0
Direct3D 9

See http://code.google.com/p/angleproject/

WebCL

- OpenCL bindings for JavaScript are coming.

N-Body Simulation
http://www.youtube.com/watch?v=F7YSQxz3j7o

http://www.khronos.org/webcl/
Prototypes for Firefox and WebKit are available

Browser Architecture

- Single Process

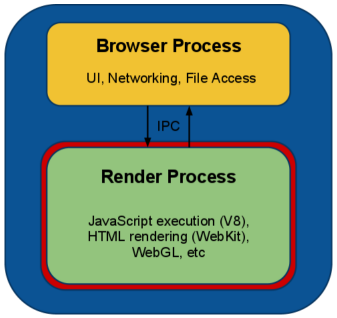
Browser Process

- UI
- Networking
- File Access
- HTML Rendering
- JavaScript execution
- Graphics
- ...

See http://www.khronos.org/assets/uploads/developers/library/2010_siggraph_bof_webgl/WebGL-BOF-2-WebGL-in-Chrome_SIGGRAPH-Jul29.pdf

Browser Architecture

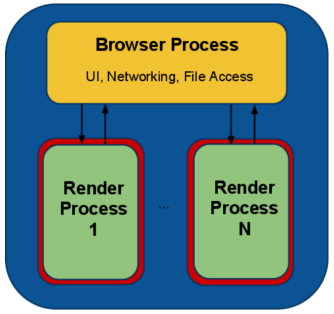
- Chrome's Multi-process



See http://www.khronos.org/assets/uploads/developers/library/2010_siggraph_bof_webgl/WebGL-BOF-2-WebGL-in-Chrome_SIGGRAPH-Jul29.pdf

Browser Architecture

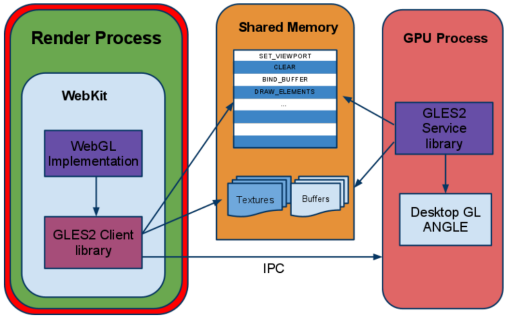
- Chrome's Multi-process



See http://www.khronos.org/assets/uploads/developers/library/2010_siggraph_bof_webgl/WebGL-BOF-2-WebGL-in-Chrome_SIGGRAPH-Jul29.pdf

Browser Architecture

- Chrome's Multi-process



See http://www.khronos.org/assets/uploads/developers/library/2010_siggraph_bof_webgl/WebGL-BOF-2-WebGL-in-Chrome_SIGGRAPH-Jul29.pdf

Questions

- In a multi-process is `gl.Get*` slow? Why?
- What about security?

Cross-Origin Resource Sharing

- Images can't always be used as texture sources. Why?

29

Cross-Origin Resource Sharing

- Same domain is OK:

```
var img = new Image();
img.onload = function() {
  gl.texImage2D(/* ... */, img);
};
img.src = "image.png";
```

30

Cross-Origin Resource Sharing

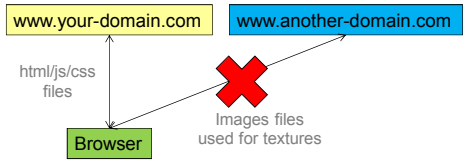
- Another domain requires *CORS* if supported:

```
var img = new Image();
img.onload = function() {
  gl.texImage2D(/* ... */, img);
};
img.crossOrigin = "anonymous";
img.src =
"http://another-domain.com/image.png";
```

31

Cross-Origin Resource Sharing

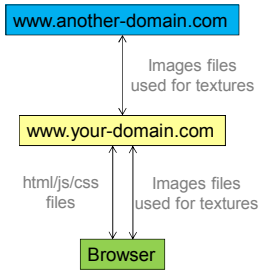
- Not all servers support CORS:



32

Cross-Origin Resource Sharing

- Use a proxy server:



"proxy.php?http://another-domain.com/image.png"

See http://resources.esri.com/help/9.3/arcgisserver/apis/javascript/arcgis/help/jshelp/ags_proxy.htm

Denial of Service Attacks

- Long draw calls
  - Complicated shaders
  - Big vertex buffers
- Solutions
  - Kill long draw calls
  - Forbid further rendering

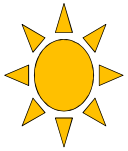
Lots of WebGL security info: http://learningwebgl.com/blog/?p=3890

WebGL Libraries

- Three.js: https://github.com/mrdoob/three.js/
- Cesium: http://cesium.agi.com/
- Many more: http://www.khronos.org/webgl/wiki/User_Contributions

WebGL Resources

- WebGL Camps: http://www.webglcamp.com
- Learning WebGL: http://learningwebgl.com



The Joys of JavaScript

Skip the next 30 slides if you already know JavaScript

JavaScript is weakly typed…

JavaScript Type System

- `short`, `int`, `float`, `double`. Who needs them?

```
var n = 1;
```


JavaScript Type System

- JavaScript has numbers, strings, and booleans:

```
var n = 1;
var s = "WebGL";
var b = true;
```

JavaScript Type System

- This compiles:

```
var n = 1;
var s = "WebGL";
var b = true;

var sum = n + s + b;
```

41

JavaScript is a functional language…

42

JavaScript Functions

- Looks familiar:

```
function add(x, y) {
  return x + y;
}

var sum = add(1, 2);
```

- Functions are first-class objects, so…

43

JavaScript Functions

- Functions are objects:

```
var add = function(x, y) {
 return x + y;
};

var sum = add(1, 2);
```

44

JavaScript Functions

- Pass functions to functions:

```
var add = function // ...

function execute(op, x, y) {
  return op(x, y);
}

var sum = execute(add, 1, 2);
```

45

JavaScript Anonymous Functions

- Why name functions?

```
function execute(op, x, y) // ...

var sum = execute(function(x, y) {
 return x + y;
}, 1, 2);
```

46

JavaScript Closures

- Why limit scope?

```
var z = 3;

var sum = execute(function(x, y) {
 return x + y + z;
}, 1, 2);
```

47

JavaScript is a dynamic language…

48

JavaScript Object Literals

- Who needs `struct`? Create objects on the fly:

```
var position = {
  x : 1.0,
  y : 2.0
};
```

49

JavaScript Object Literals

- Why not add fields on the fly too?

```
var position = {
  x : 1.0,
  y : 2.0
};
position.z = 3.0;
```

50

JavaScript Object Literals

- Who needs `class`?

51

JavaScript Object Literals

- Who needs `class`? Create functions too:

```
var position = {
  x : 1.0,
  y : 2.0,
  min : function() {
    return Math.min(this.x, this.y);
  }
};
```

52

JavaScript Object Literals

- Why not change `min()`?

```
position.z = 3.0;
position.min = function() {
  return Math.min(this.x, this.y,
    this.z);
};
```

53

JavaScript Object Literals

- Useful for passing to functions. Why?

54

JavaScript Object Literals

- Useful for passing to functions. Why?
- What do these arguments mean?

```
pick(322, 40, 5, 4);
```

55

JavaScript Object Literals

- Useful for passing to functions. Why?
- What do these arguments mean?

```
pick({
  x : 322,
  y : 40,
  width : 5,
  height : 4
});
```

56

JavaScript does object-oriented…

57

JavaScript Constructor Functions

```
function Vector(x, y) {
  this.x = x;
  this.y = y;
}

var v = new Vector(1, 2);
```

58

JavaScript Constructor Functions

- Objects can have functions:

```
function Vector(x, y) {
  this.x = x;
  this.y = y;
  this.min = function() {
    return Math.min(this.x, this.y);
  };
}
```

59

JavaScript Constructor Functions

- Objects have prototypes:

```
function Vector(x, y) {
  this.x = x;
  this.y = y;
}

Vector.prototype.min = function() {
  return Math.min(this.x, this.y);
};
```

60

JavaScript Polymorphism

- No need for virtual functions

```
function draw(model) {
  model.setRenderState();
  model.render();
}
```

61

JavaScript Polymorphism

- No need for virtual functions

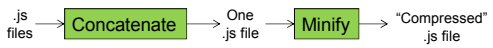
```
var level = {
  setRenderState : function() // ...
  render : function() // ...
};

draw(level); // Just works
```

62

JavaScript Build Pipeline

- Different than C++
- *Goal*: fast downloads
- Common:



- Alternative: fine-grain modules
- How do you deploy shaders?

63

See http://www.julienlecomte.net/blog/2007/09/16/

JavaScript Advice

- Use JSHint
- Have excellent test coverage
- Use the Chrome and Firefox debuggers

64

JavaScript Resources

I promise I do not work for O'Reilly or Yahoo