DTU

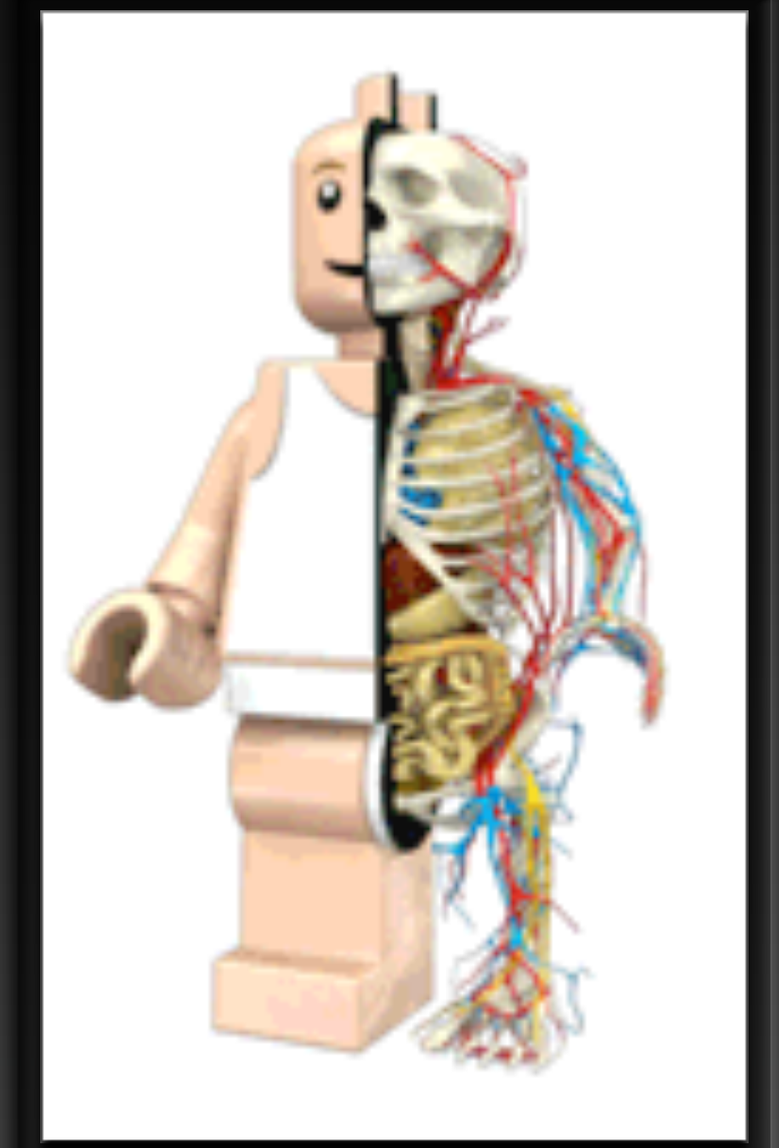
WebGL overview

Morten Nobel-Jørgensen

PhD Student

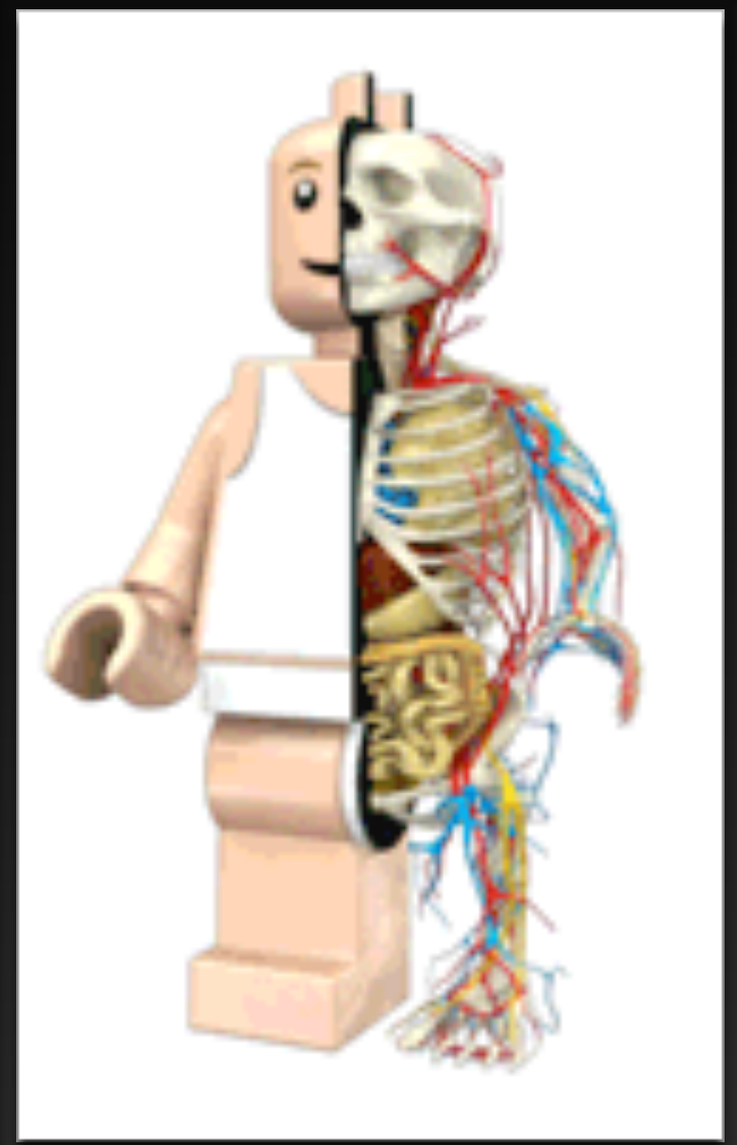
Agenda

- Overview of WebGL
- WebGL performance
- WebGL architecture
- Future predictions



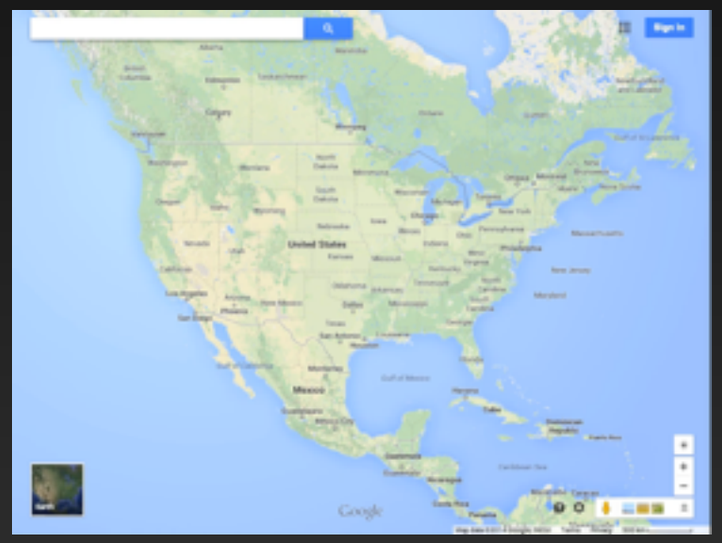
Agenda

- Overview of WebGL
- WebGL performance
- WebGL architecture
- Future predictions



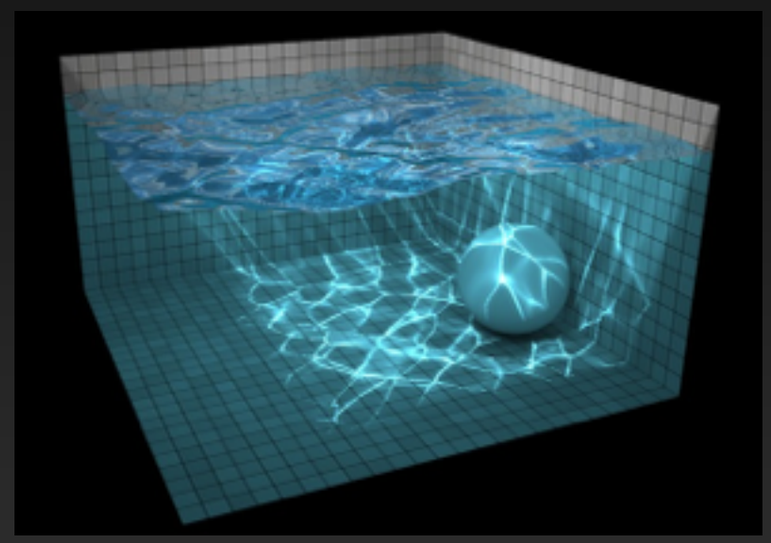
Overview of WebGL

WebGL Usages

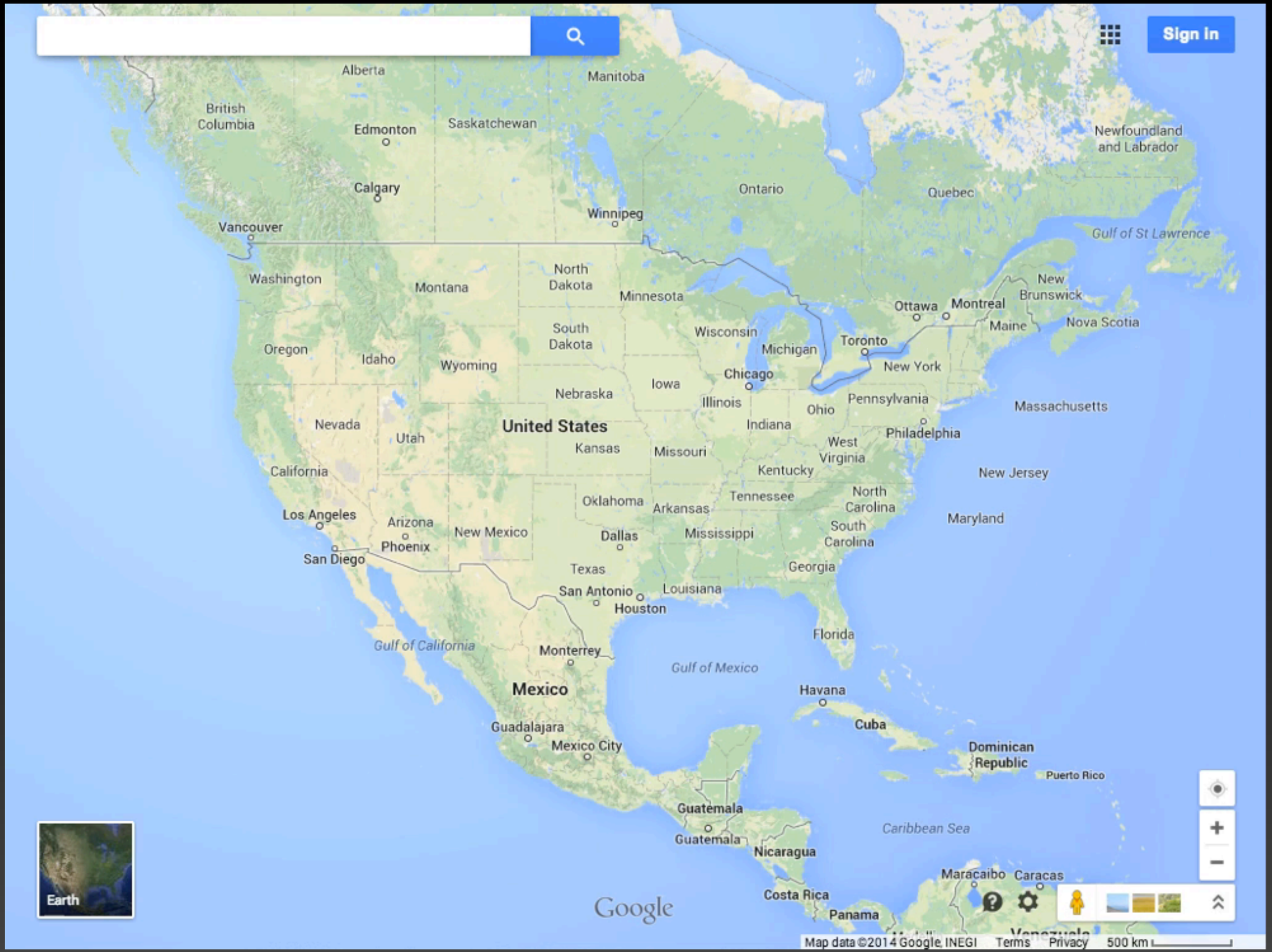


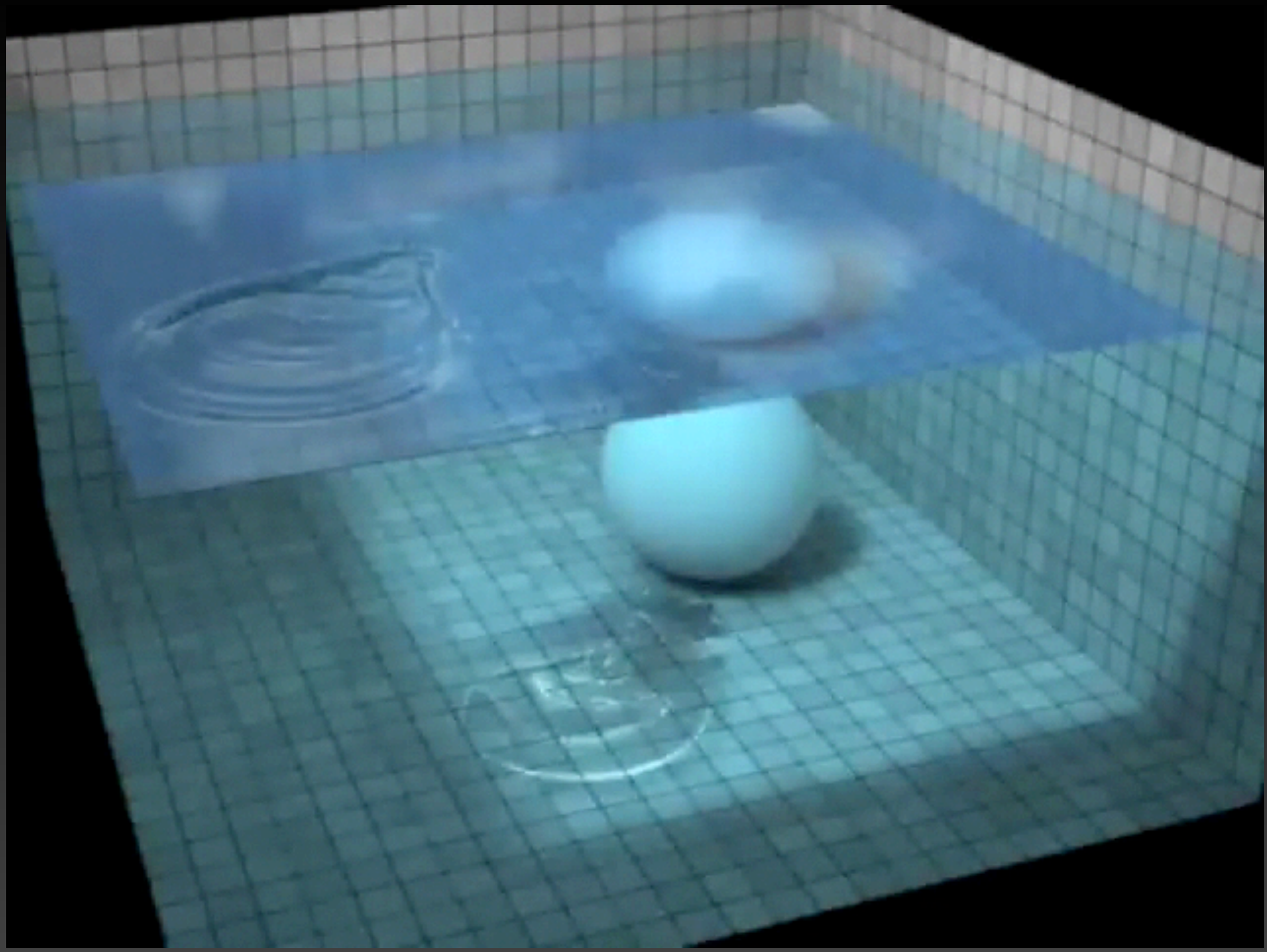
Google Maps

Ro.Me



GLWater


Sign In
Alberta
Manitoba
British Columbia
Edmonton
Saskatchewan
Newfoundland and Labrador
Calgary
Ontario
Quebec
Winnipeg
Vancouver
Gulf of St Lawrence
Washington
North Dakota
Montana
Minnesota
New Brunswick
Ottawa
Montreal
South Dakota
Wisconsin
Maine
Nova Scotia
Oregon
Idaho
Michigan
Toronto
Wyoming
New York
Chicago
Iowa
Nebraska
Illinois
Pennsylvania
Ohio
Massachusetts
Nevada
United States
Indiana
Philadelphia
Utah
Kansas
Missouri
West Virginia
California
Kentucky
New Jersey
Oklahoma
Arkansas
Tennessee
North Carolina
Los Angeles
Arizona
New Mexico
Dallas
Mississippi
Maryland
South Carolina
Phoenix
San Diego
Texas
Georgia
San Antonio
Louisiana
Houston
Florida
Gulf of California
Monterrey
Gulf of Mexico
Mexico
Havana
Cuba
Guadalajara
Mexico City
Dominican Republic
Puerto Rico
Guatemala
Caribbean Sea
Guatemala
Nicaragua
Maracaibo
Caracas
Costa Rica
Panama
Earth
Google
Map data ©2014 Google, INEGI
Terms
Privacy
500 km



ROME "3 Dragons Of Black"

What is WebGL

- 3D graphics API
- Hardware accelerated
- JavaScript
- For web-browsers

Why WebGL

- 3D content in browser
- Plugin free
- Secure
- Utilize the hardware

WebGL Family Tree

Fixed function

OpenGL 1.5

Shader based

OpenGL 2.0

WebGL Family Tree

Fixed function

Shader based

OpenGL 1.5 → OpenGL 2.0

OpenGL ES 1.0 → OpenGL ES 2.0

WebGL Family Tree

Fixed function

Shader based

OpenGL 1.5 → OpenGL 2.0

OpenGL ES 1.0 → OpenGL ES 2.0

WebGL 1.0

WebGL Family Tree

Fixed function

Shader based

Recent

OpenGL 1.5

OpenGL 2.0

4.x

OpenGL ES 1.0

OpenGL ES 2.0

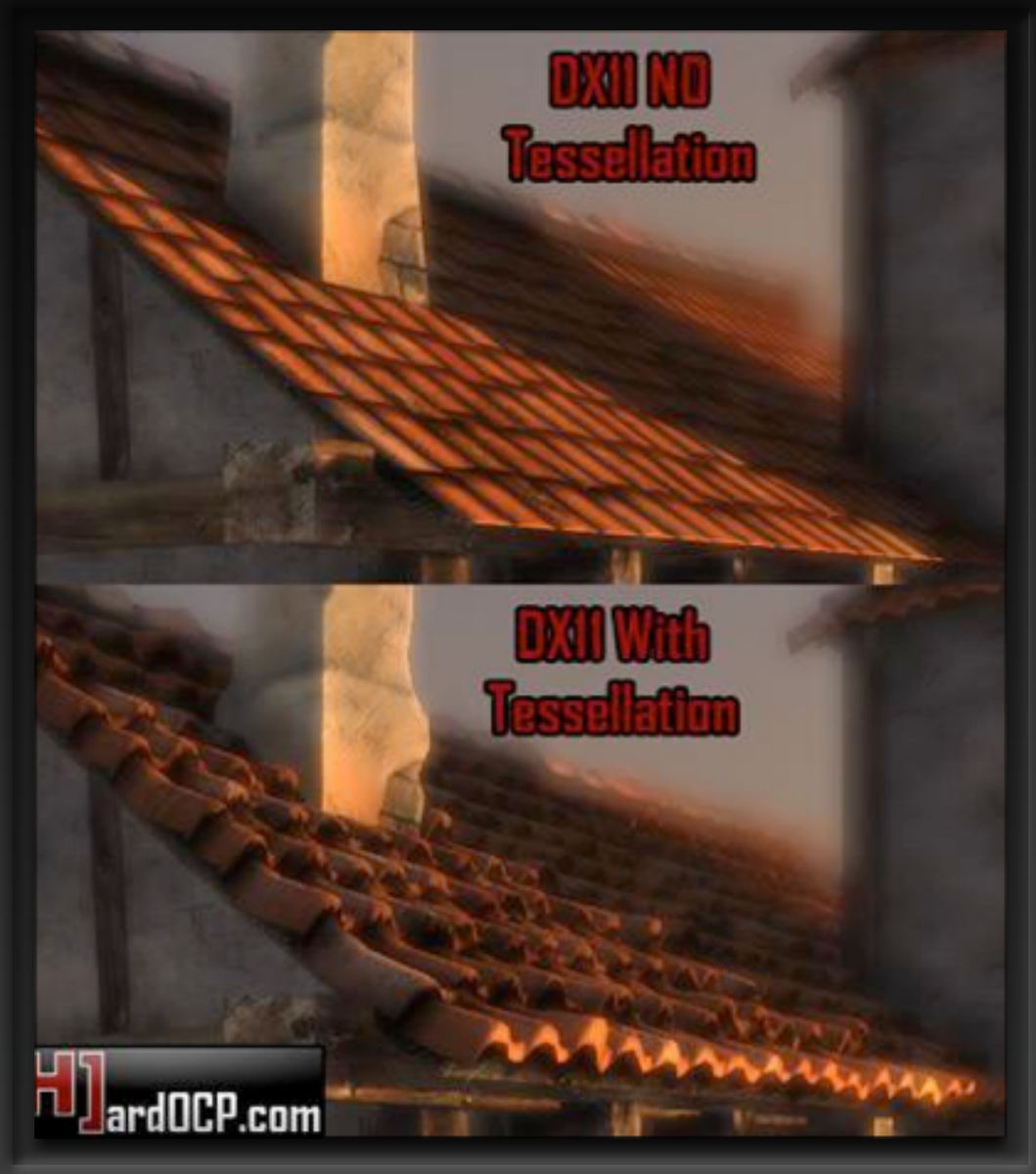
3.x

WebGL 1.0

2.x

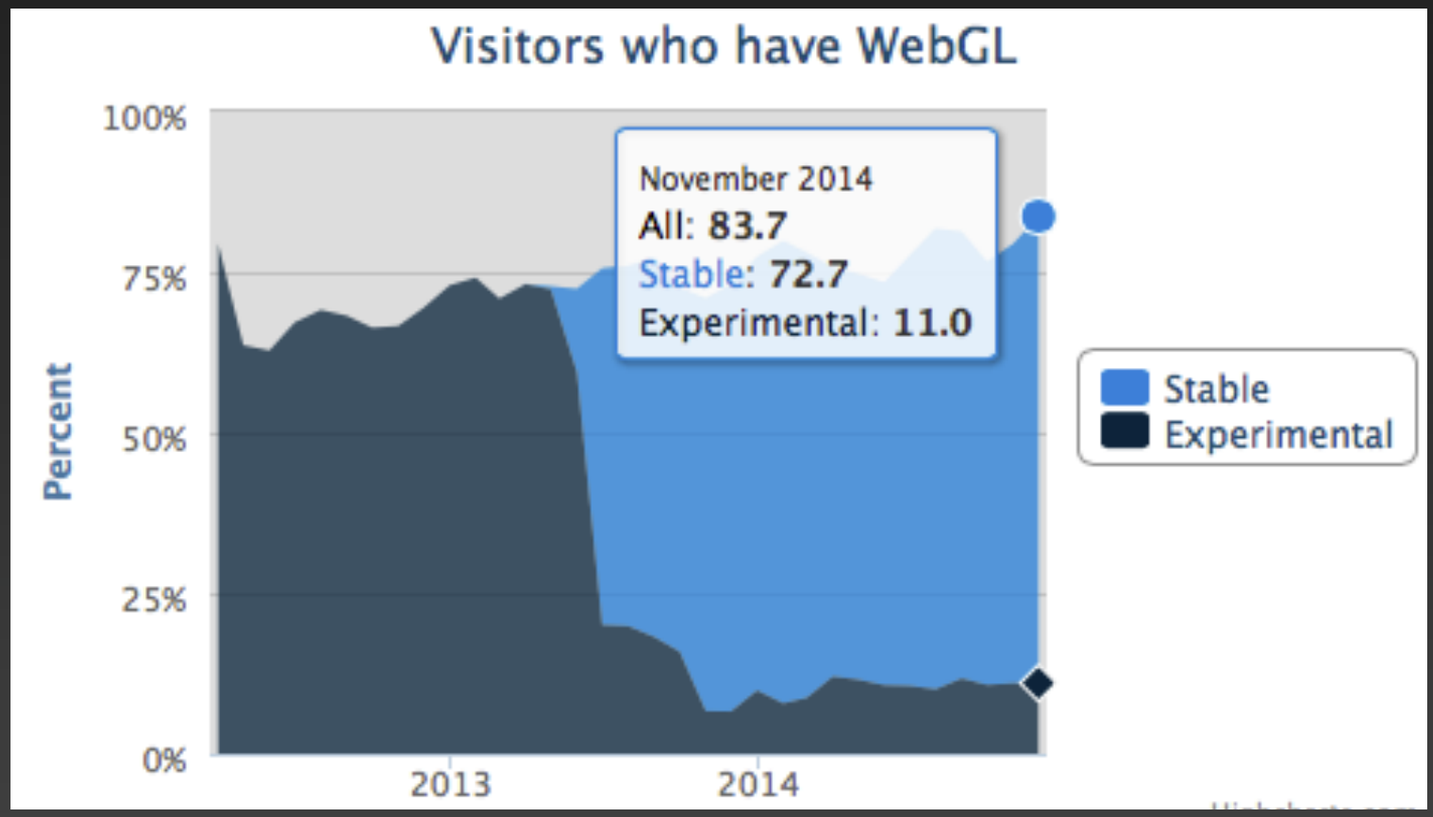
Limitations of WebGL

- No multiple render targets (MRT)
- No geometry shaders
- No tessellation shaders
- No immediate mode rendering



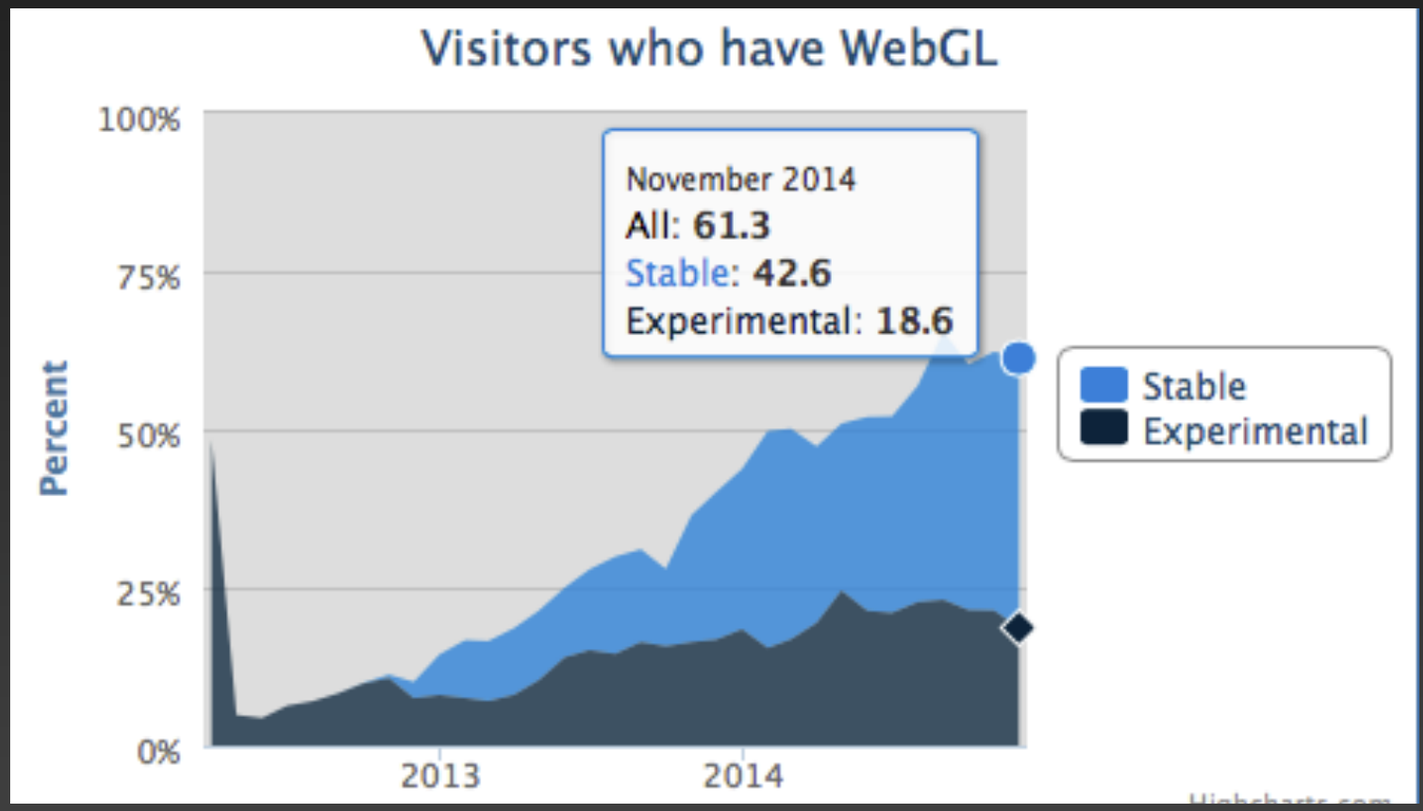
WebGL Browser Support

- Personal computers: **84 %**



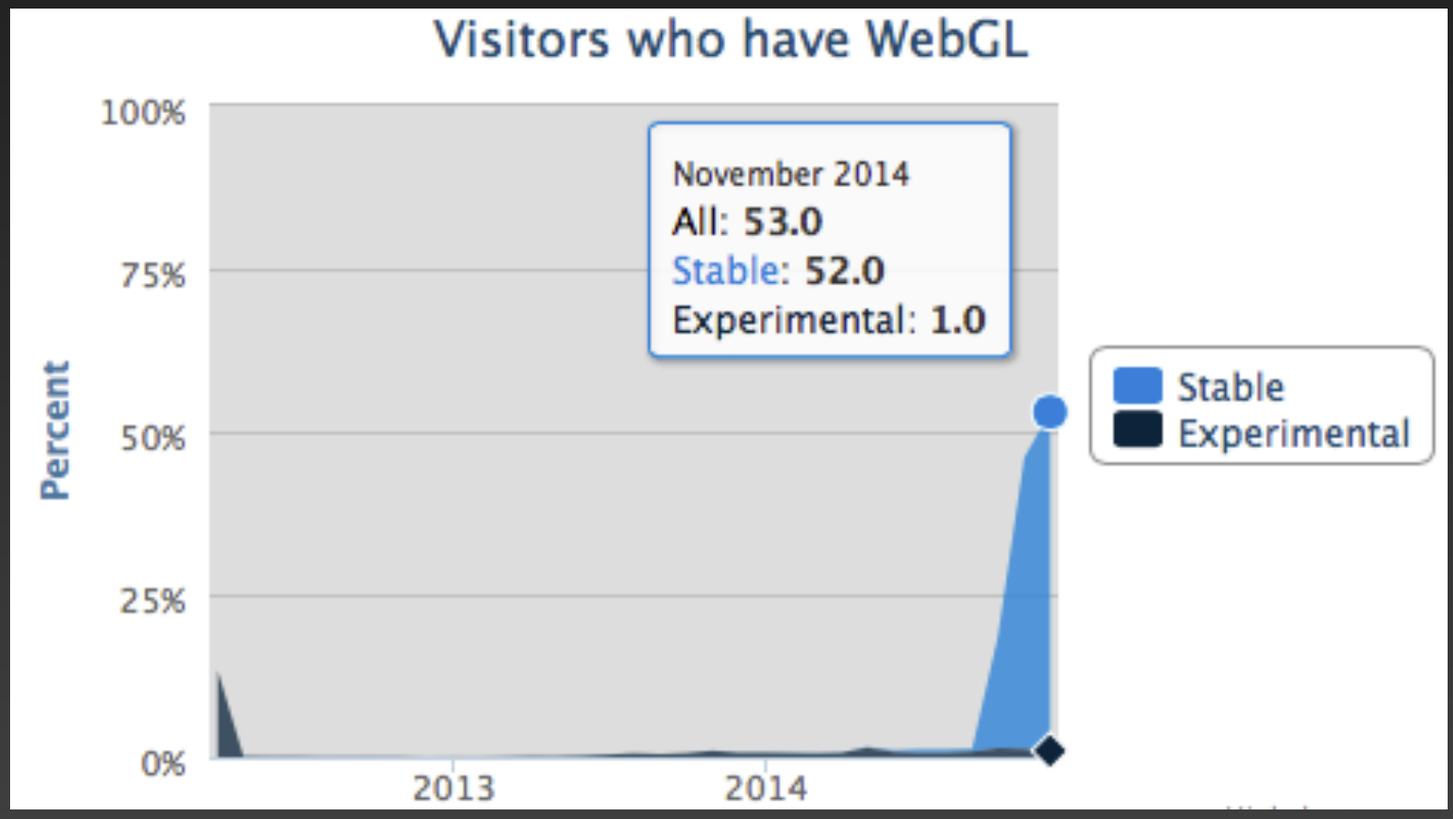
WebGL Browser Support

- Personal computers: **84 %**
- Android devices: **61 %**



WebGL Browser Support

- Personal computers: **84 %**
- Android devices: **61 %**
- iOS devices: **53%**



WebGL 1.0 API

- C like API
- Flat structure
- 135 functions
- 296 “enums”

WebGL program structure

Create Context

Init resources

Render loop

WebGL program structure

Create Context

- Create WebGL Canvas in HTML
- Object WebGL reference

Init resources

Render loop

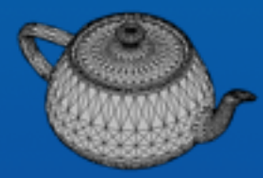
WebGL program structure

Create Context

Init resources

Render loop

- Load Resources
  - Textures
  - Shaders
  - Meshes
- Upload to GPU



WebGL program structure

Create Context

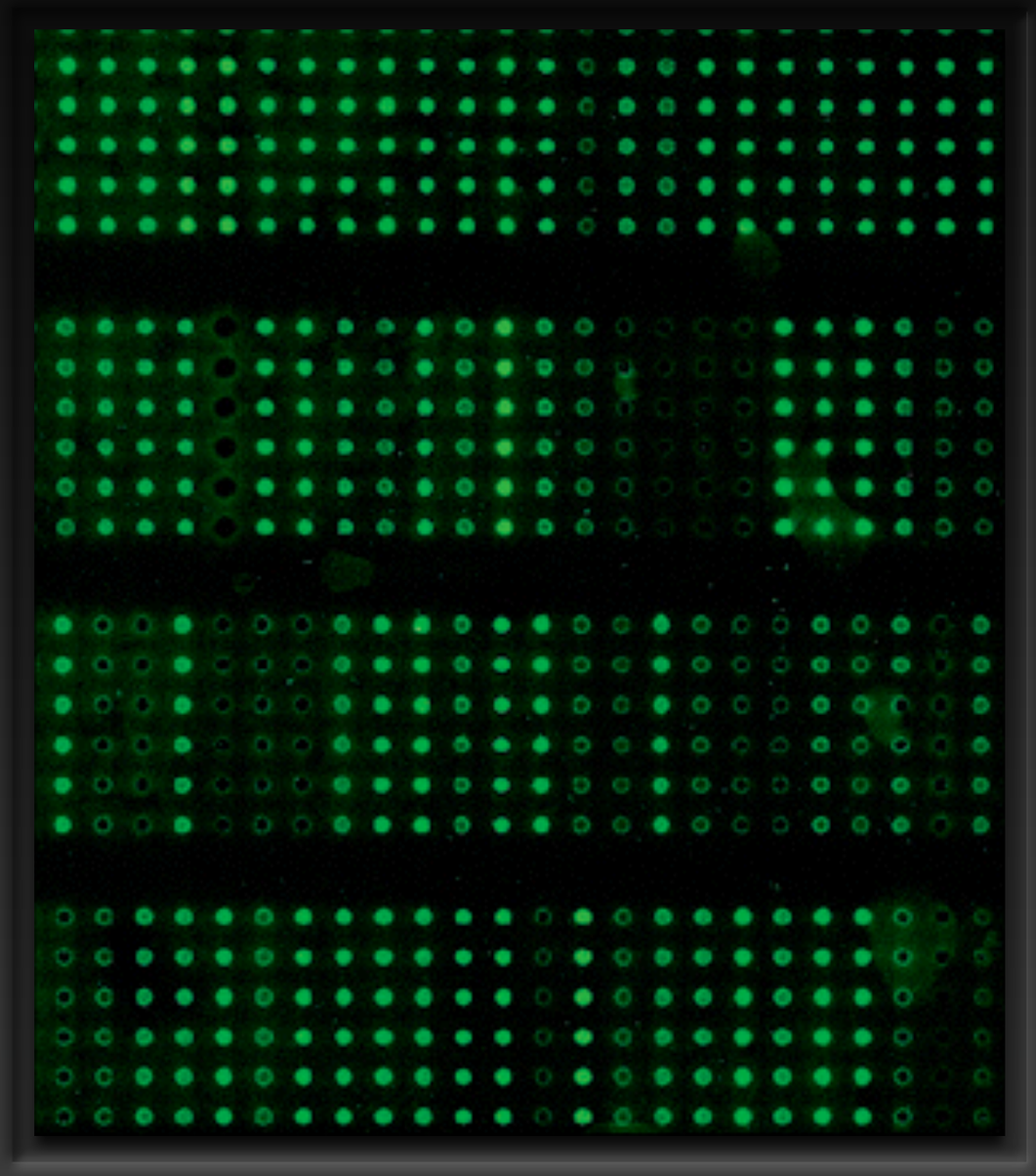
Init resources

Render loop

- Set camera
- For each model
  - Set shader
  - Render model

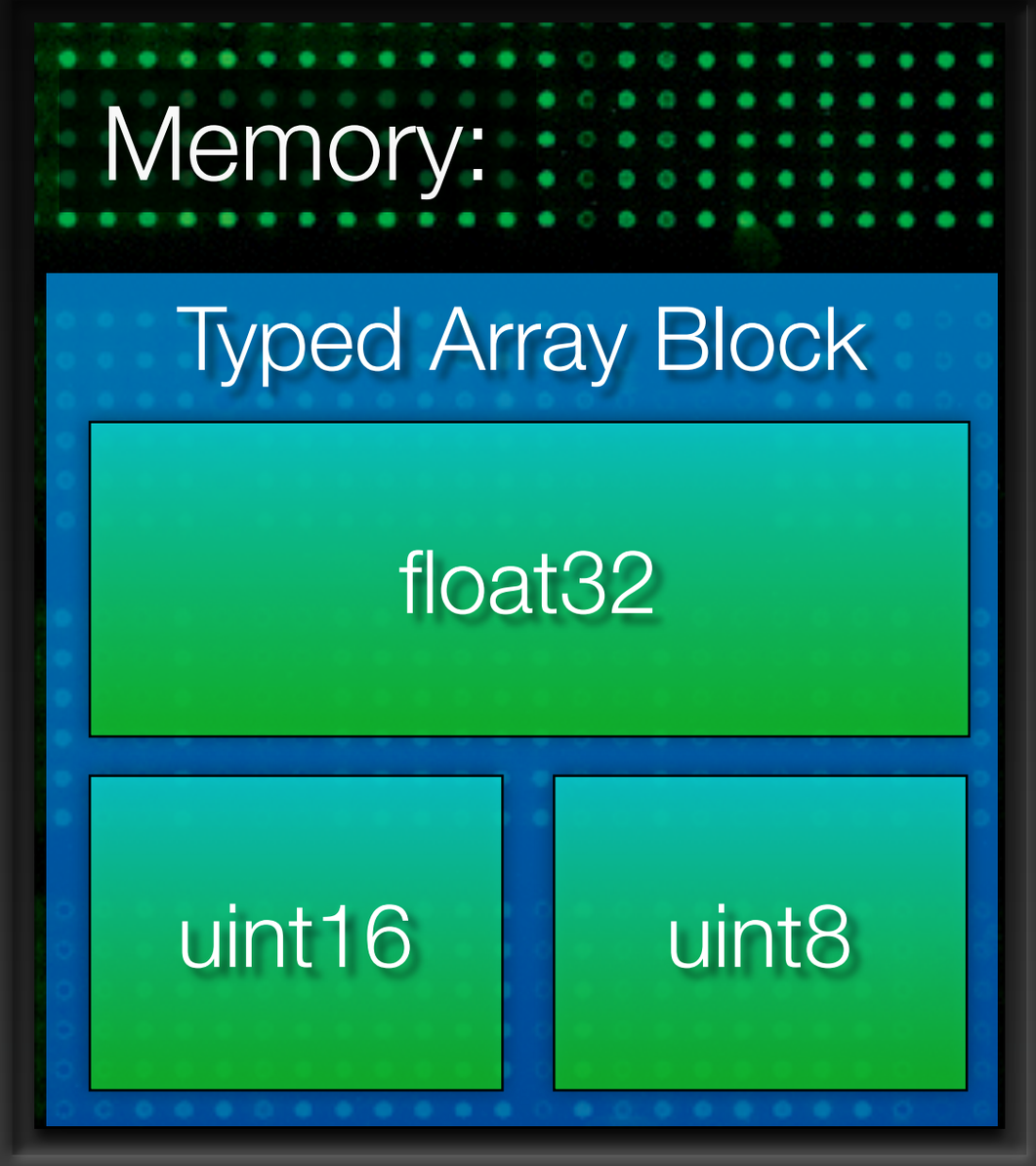
JavaScript and binary data

- Problem:
  - Uploading data to GPU must be done in binary format
  - JavaScript only has 64 bit floating point



JavaScript and binary data

- Problem:
  - Uploading data to GPU must be done in binary format
  - JavaScript only has 64 bit floating point
- Solution: Typed Array API
  - Creation of binary arrays
  - Typed views on binary arrays (float32, uint16, etc.)



WebGL security

- No read from uninitialized textures
- No use of cross-origin content without permission
- Out of range memory access check
- Blacklist of poor drivers

Resource management

- Manuel deallocation
- Uses handles
- Clean up on page close

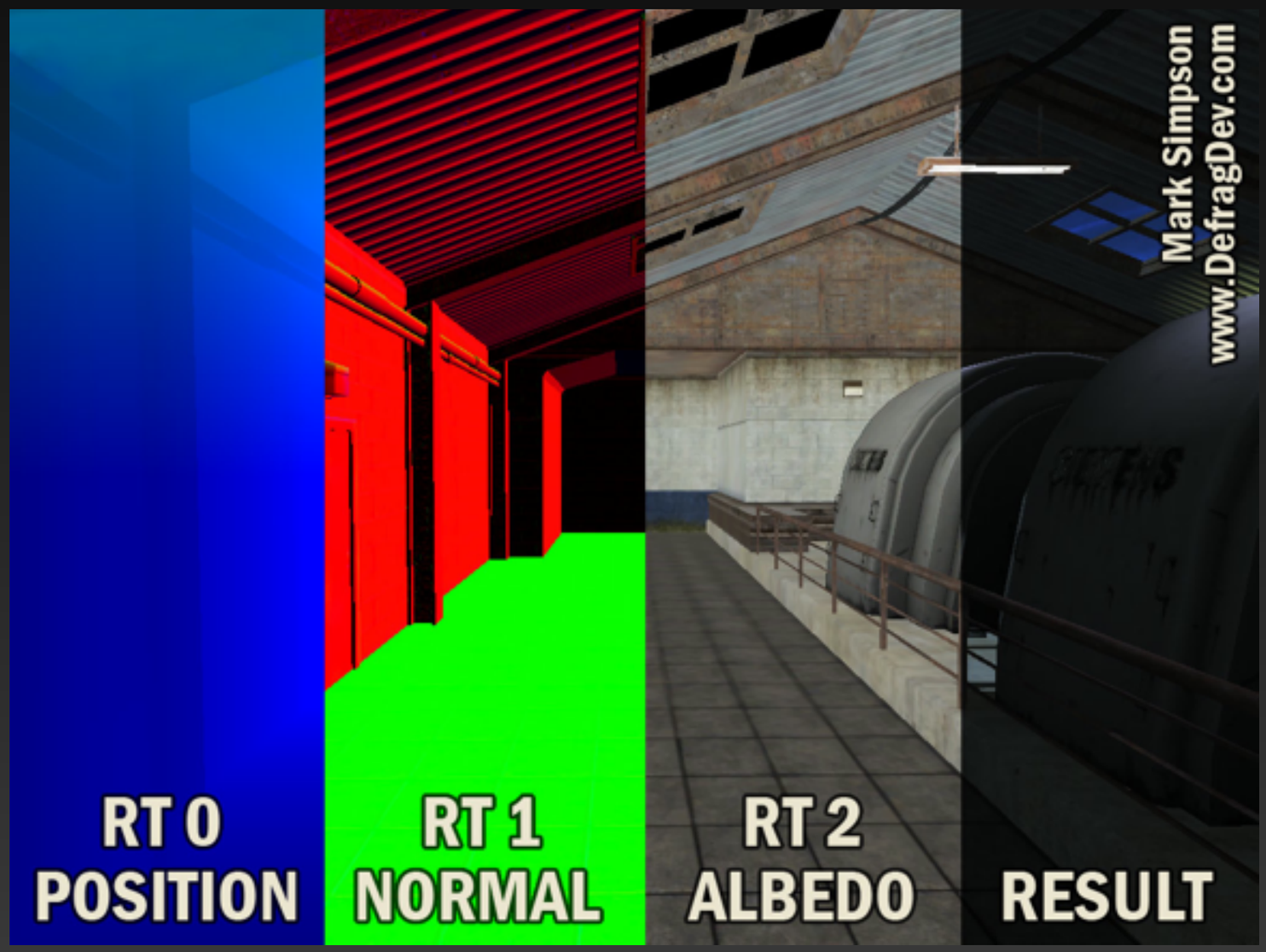
New features in WebGL 2.0

Multiple Render Targets (MRT)

Allows shaders to write to multiple targets
Instead of just rendering directly to the screen

New in WebGL 2.0

Multiple Render Targets (MRT)



Deferred Rendering

New in WebGL 2.0

Multiple Render Targets (MRT)

Screen Space Ambient Occlusion

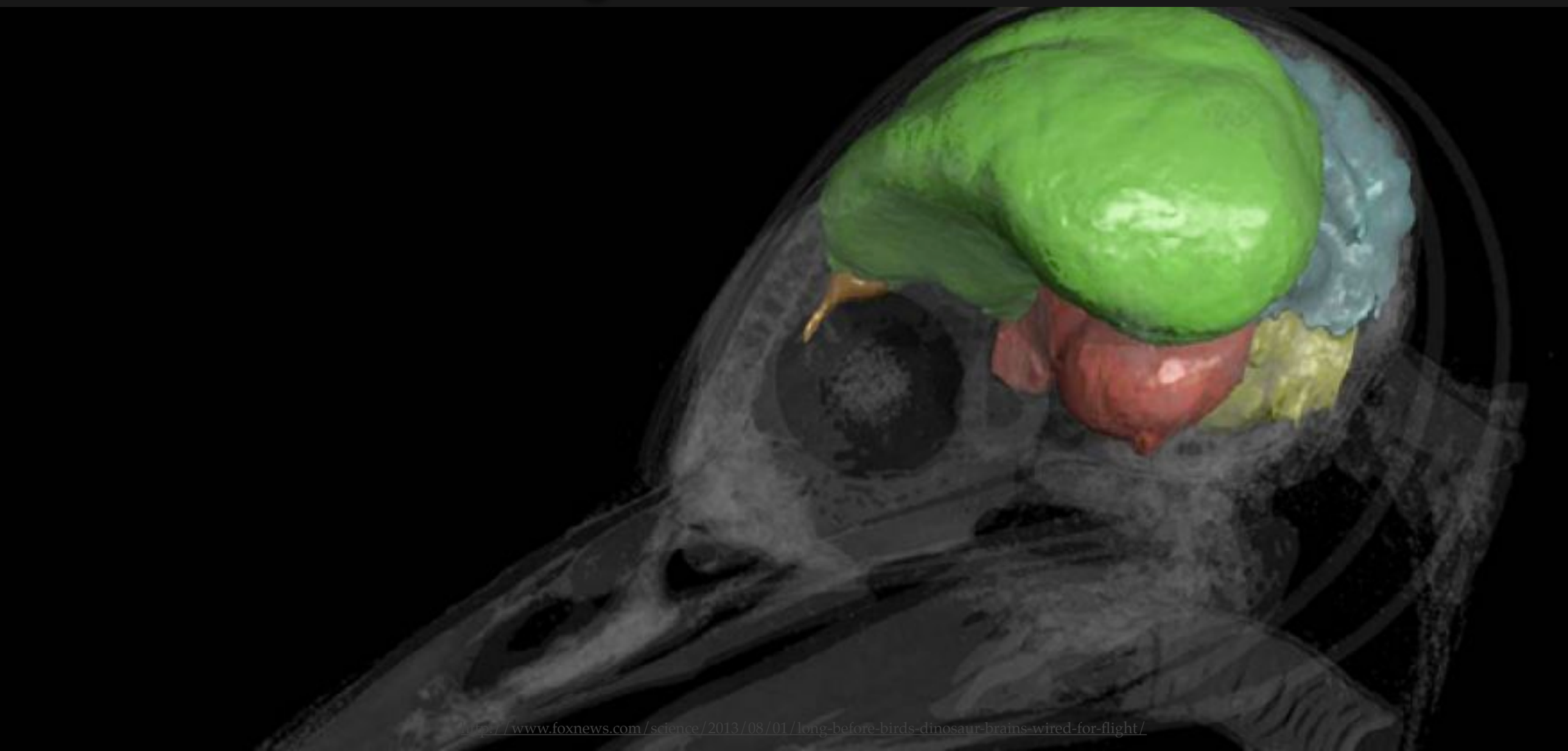
http://www.moddb.com/engines/3d-game-studio/images/ssao-a7

Multiple Render Targets (MRT)

Depth Of Field

3D textures

- Great for visualising volumetric data



http://www.foxnews.com/science/2013/08/01/long-before-birds-dinosaur-brains-wired-for-flight/

ETC2 / EAC texture compression

- Previously no standardised texture format and was optional
- Mandatory in WebGL 2.0
- Compression for
  - 1-3 color channels
  - 0-1 alpha channel

Other stuff in WebGL 2.0

Transform Feedback

Query Objects

Instancing

Fragment Depth

Vertex Array Objects

Multisampled Renderbuffers

Sampler Objects

Sync Objects

Uniform Buffer Objects

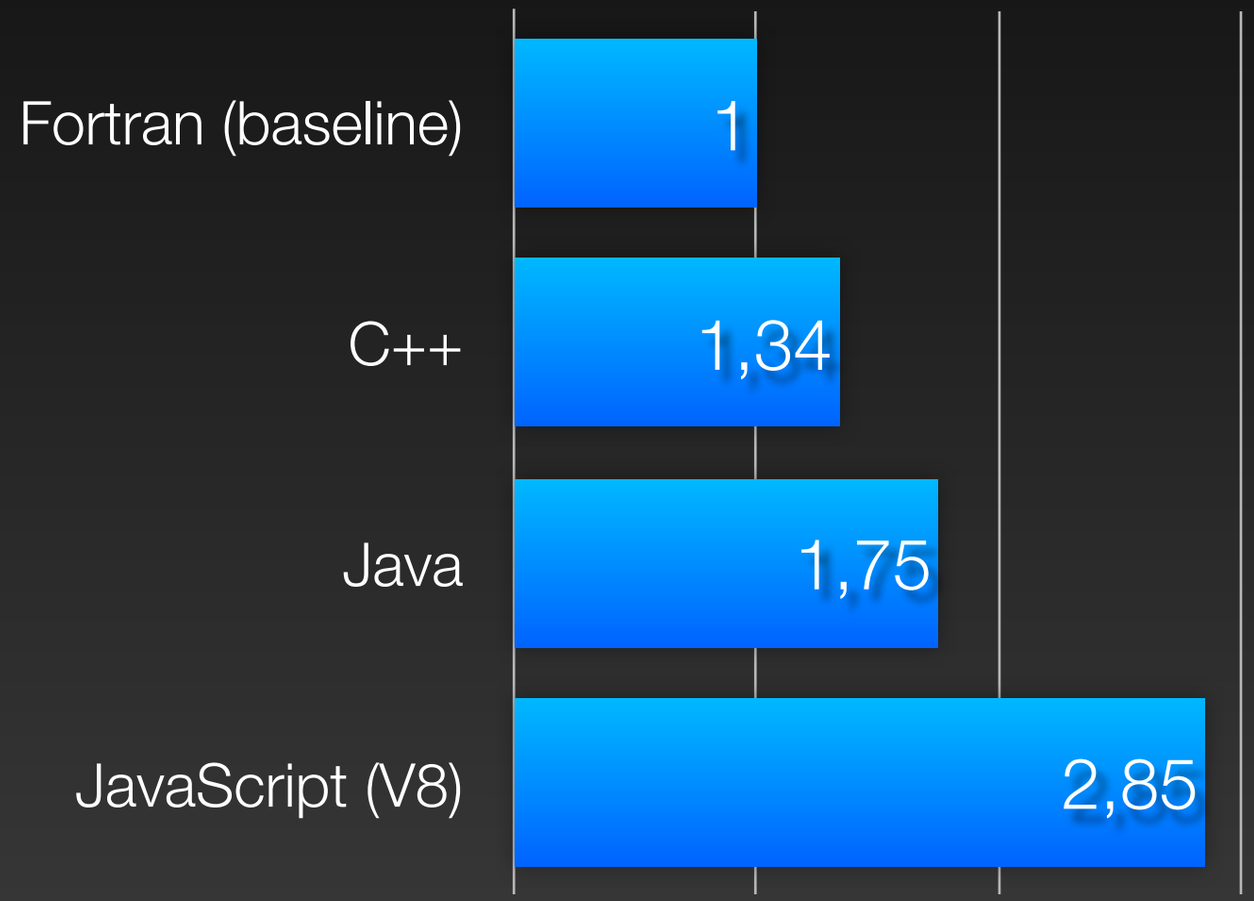
WebGL performance

Performance of JavaScript

- Data from: The Computer Language Benchmarks Game (shootout.alioth.debian.org)
- Warning: The results may reflect real world applications

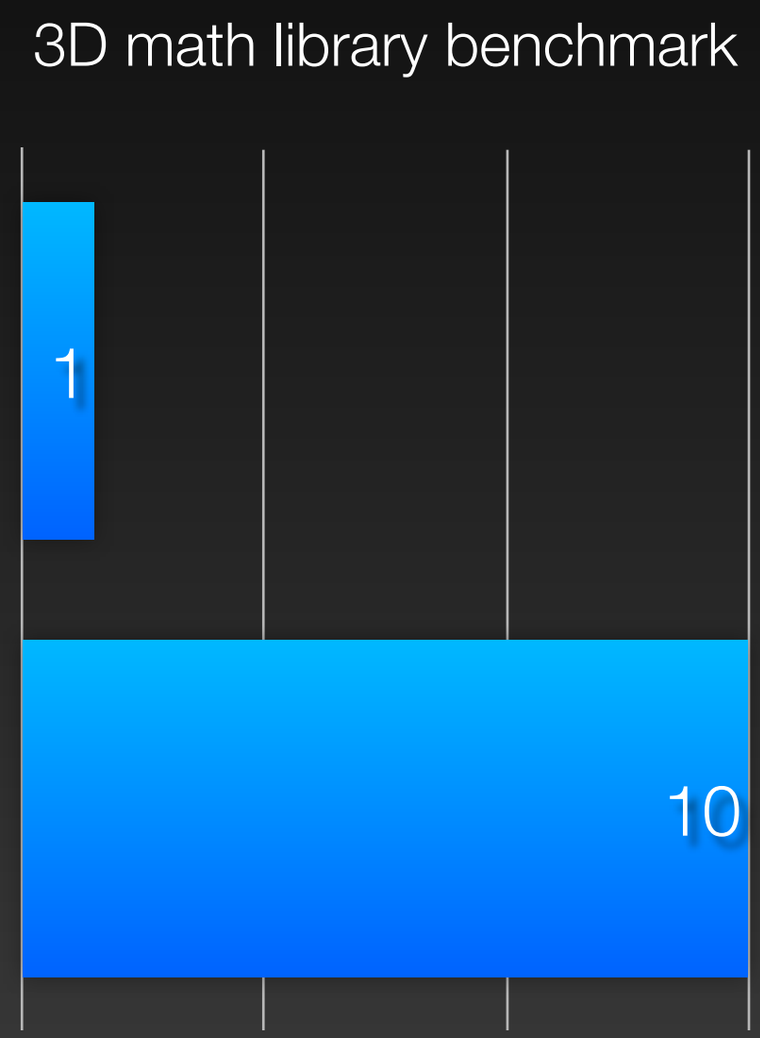
Performance of JavaScript

- Data from: The Computer Language Benchmarks Game
(shootout.alioth.debian.org)
- Warning: The results may reflect real world applications



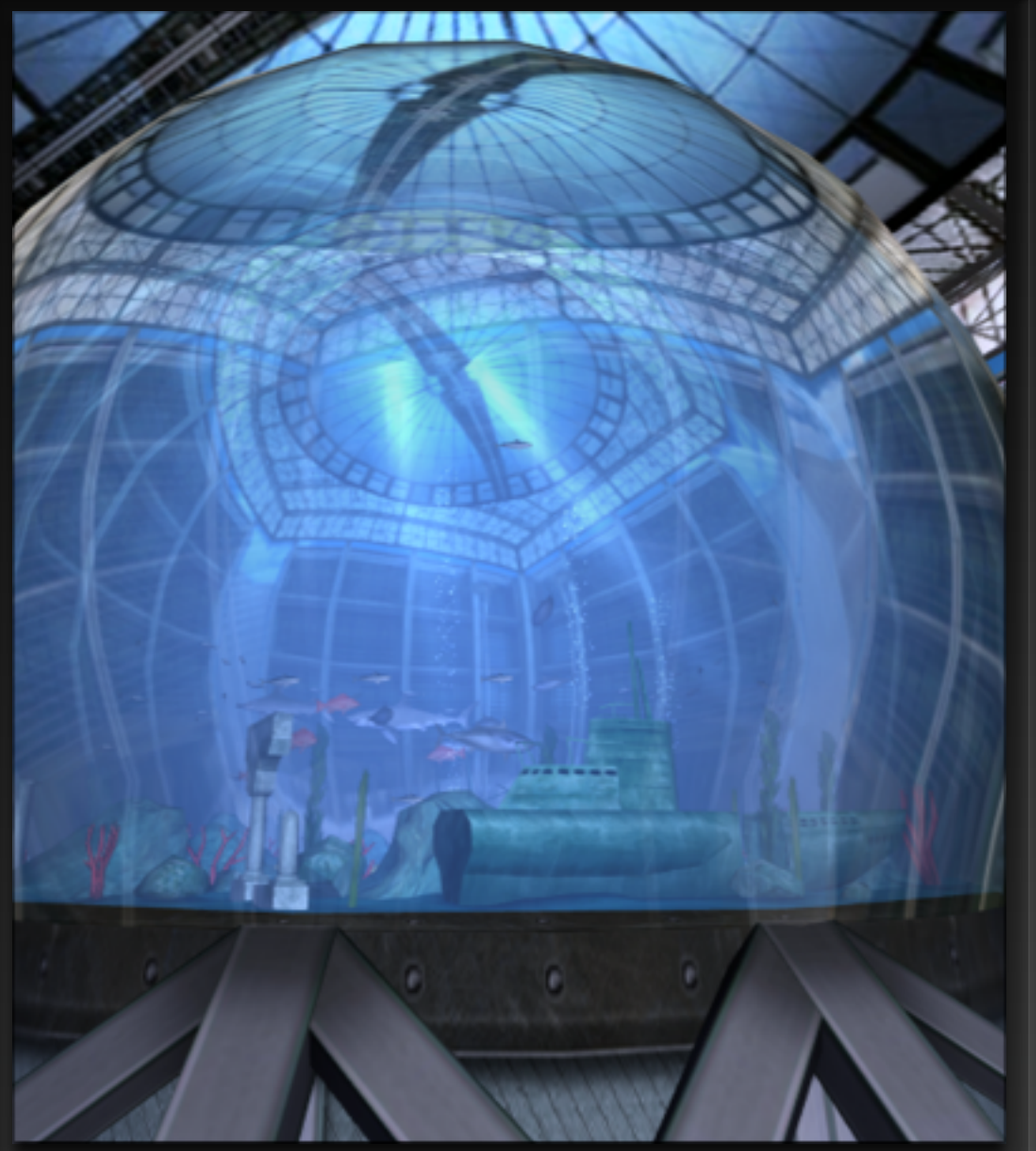
Performance of JavaScript

- Data from: The Computer Language Benchmarks Game (shootout.alioth.debian.org)
- Warning: The results may reflect real world applications
- For 3D math: Around 10 times slower than C++



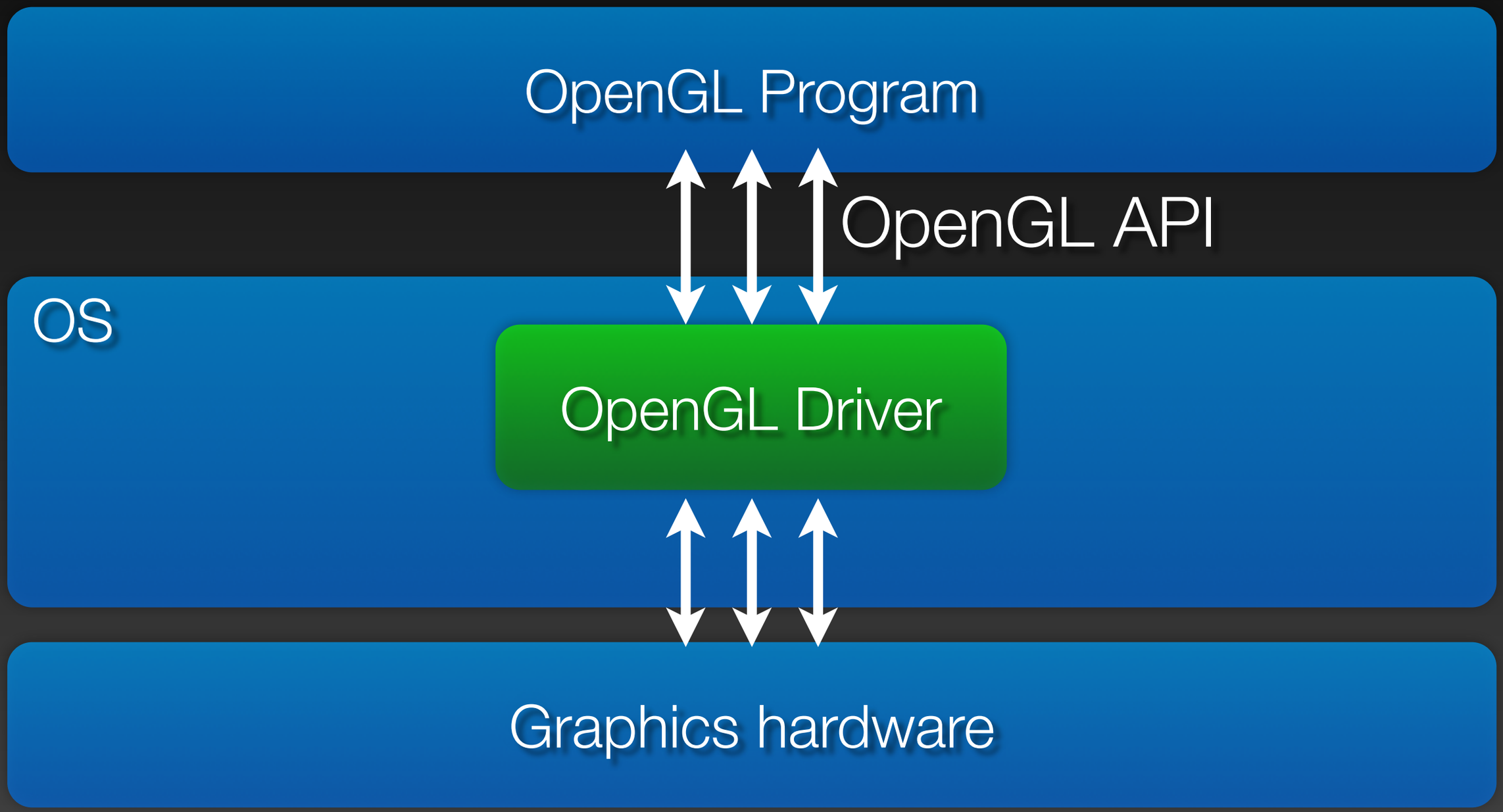
C++ and Java beats JavaScript because ...

- JavaScript is **dynamically typed**
- JavaScript has **high level of indirectness**
- JavaScript is compiled **on-the-fly**

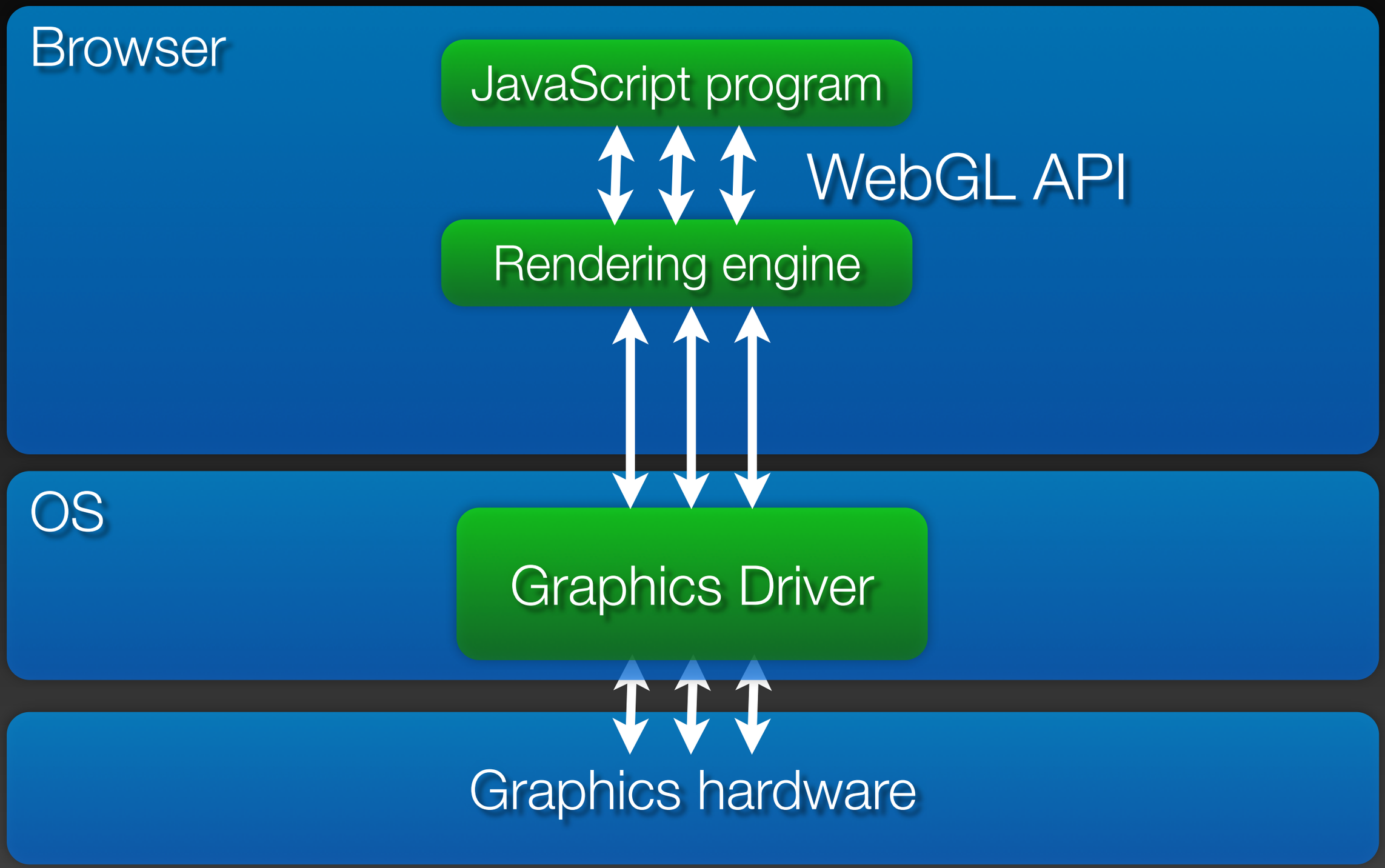


WebGL architecture

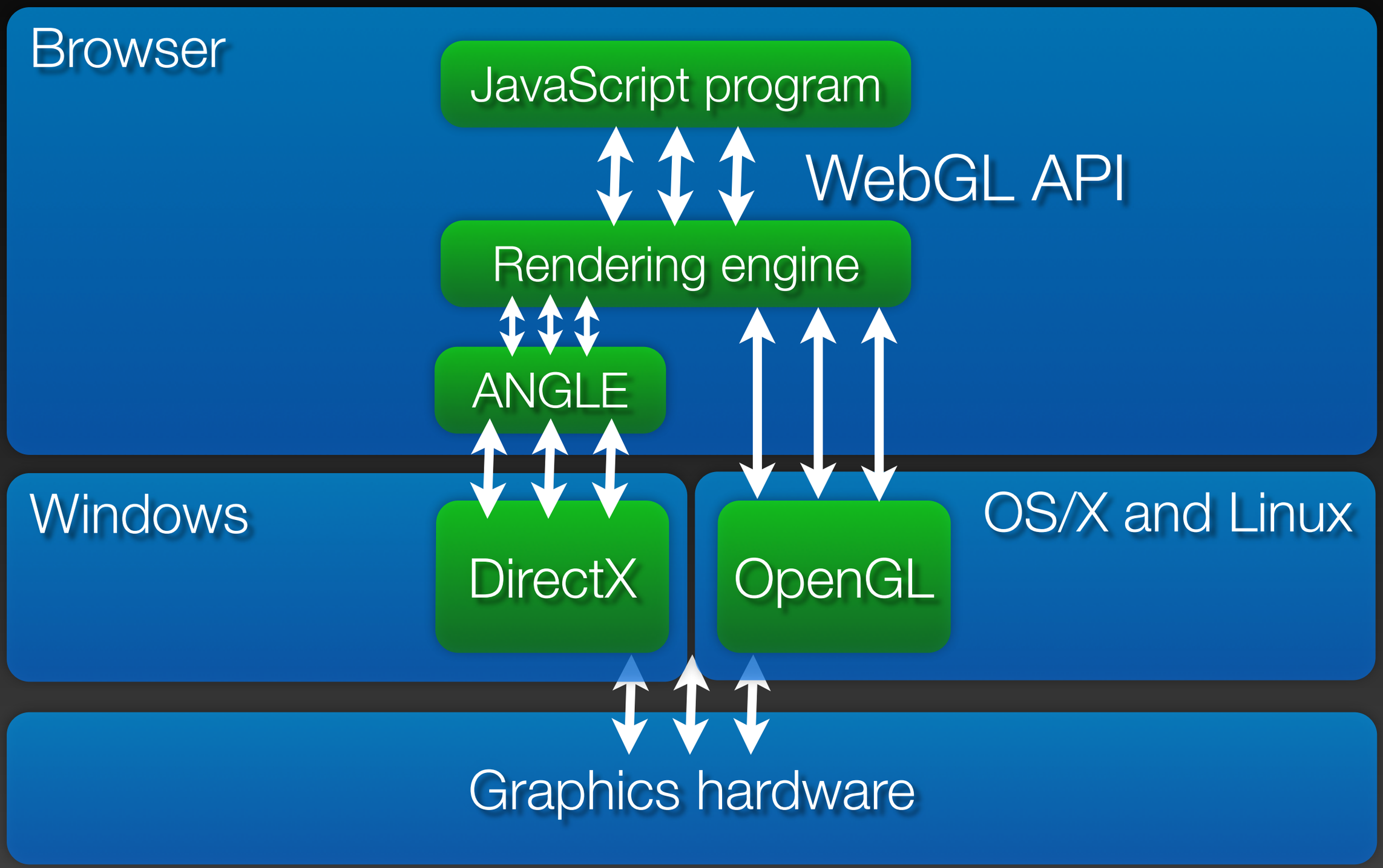
Desktop OpenGL architecture



JavaScript WebGL architecture



JavaScript WebGL architecture



References

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

- http://shootout.alioth.debian.org/u32/which-programming-languages-are-fastest.php