

Overview of Javascript in Games

by
Javier Arevalo

So, who am I?

- Making commercial games since 1986
- On personal computers, consoles, social/web, mobile
- Programmer, designer, producer, wannabee artist and sound dude
- Graphics technology, core engine, user interface, AI, backend
- Used many programming languages, from assembly to Javascript



Wait, Javascript?

- Most 'AAA' PC/console games are written in C++
- Flash dominates the web and social games scene
- Objective-C, Java and C# rule their respective mobile platforms
- Servers in Java, PHP, Python, Ruby, Go...

Makes sense to prefer native code. Games often require:

- Raw performance
- Real-time execution
- Complex / flexible input
- Rich visuals and audio
- Storage, networking



Hardcore Game Programmers



But, but, but

- Javascript is not compiled
- Javascript is garbage collected
- Javascript is dynamically typed
- Javascript has no decent library support
- Javascript is not modular, not OOP, not safe, just good for *scripts*!
- Javascript is not suitable for large projects!
- Javascript is an afterthought!
- Javascript doesn't require semicolons!
 - (mostly)
 - (depends on who you ask)
 - (there be dragons there)

Javascripters in action



So, who made it possible?

HTML5



PhoneGap



COCOON^{JS}



and, in my opinion...



- *Apple has adopted HTML5, CSS and JavaScript*
- *Flash is the number one reason Macs crash*
- *Flash has not performed well on mobile devices*
- *Flash was designed for PCs using mice*
- *Letting a third party layer of software come between the platform and the developer ultimately results in sub-standard apps*
 - *Oh wait...*

<http://www.apple.com/hotnews/thoughts-on-flash/>

How do we make games with Javascript?

Text-based games

- Function pretty much like a standard web site
- Common in many early RPG/card/strategy web games
- Interaction limited to clicking buttons or regions
- Can only do 2D with pre-defined graphics
- Little or not animation or sound
- Mafia Wars: Javascript / PHP
 - Single-page with complex UI
 - Not small: 34k LOC CSS!
- Viable approach, but very limited



DOM-based games

- Use divs and advanced CSS to display and animate graphics
- Lots of absolute positioning and CSS transforms
- Heavy use of sprites and static backgrounds
- Easy to take advantage of hardware acceleration
- Usually limited to flat 2D
- Possible to do fake 3D, but hard to get right and compatible
- Natural fit with DOM-based UIs
- Tyrian art by Dan Cook @danctheduck



Extreme div usage: Wolf 3D

- Raycasting engine
- Thin scaled divs
- Hundreds of them!
- But all hw-accelerated



```
><div style="display: none; position: absolute; width: 128px; height: 128px;
class="sprite">...</div>
><div style="display: none; position: absolute; width: 128px; height: 128px;
class="sprite">...</div>
><div style="display: none; position: absolute; width: 128px; height: 128px;
class="sprite">...</div>
><div style="display: none; position: absolute; width: 128px; height: 128px;
class="sprite">...</div>
><div style="display: none; position: absolute; width: 128px; height: 128px;
class="sprite">...</div>
▼<div style="position: absolute; overflow: hidden; z-index: 3823740; display:
height: 70px; left: 32.294167883890054px; top: 117px; " class="sprite">
  <div style="position: absolute; display: block; top: 0px; background-size:
url(http://zmi-cdnstatic.s3.amazonaws.com/wolfenstein.bethsoft.com/art
width: 3430px; height: 70px; left: -3150px; background-repeat: no-repe
</div>
▼<div style="position: absolute; width: 3px; left: 0px; overflow: hidden;
72px; top: 116px; ">
  <div style="position: absolute; display: block; width: 192px; backgrou
image: url(http://zmi-cdnstatic.s3.amazonaws.com/wolfenstein.bethsoft.
walls.png); height: 8640px; top: -72px; left: -84px; "></div>
</div>
▶<div style="position: absolute; width: 3px; left: 3px; overflow: hidden;
71px; top: 116.5px; ">...</div>
▶<div style="position: absolute; width: 3px; left: 6px; overflow: hidden;
71px; top: 116.5px; ">...</div>
▶<div style="position: absolute; width: 3px; left: 9px; overflow: hidden;
71px; top: 116.5px; ">...</div>
▶<div style="position: absolute; width: 3px; left: 12px; overflow: hidden;
70px; top: 117px; ">...</div>
▶<div style="position: absolute; width: 3px; left: 15px; overflow: hidden;
127px; top: 88.5px; ">...</div>
▶<div style="position: absolute; width: 3px; left: 18px; overflow: hidden;
129px; top: 87.5px; ">...</div>
▶<div style="position: absolute; width: 3px; left: 21px; overflow: hidden;
132px; top: 86px; ">...</div>
▶<div style="position: absolute; width: 3px; left: 24px; overflow: hidden;
135px; top: 84.5px; ">...</div>
▶<div style="position: absolute; width: 3px; left: 27px; overflow: hidden;
137px; top: 83.5px; ">...</div>
▶<div style="position: absolute; width: 3px; left: 30px; overflow: hidden;
139px; top: 82.5px; ">...</div>
```

FLOOR	SCORE	LIVES		HEALTH	AMMO
3	100	3		29%	10

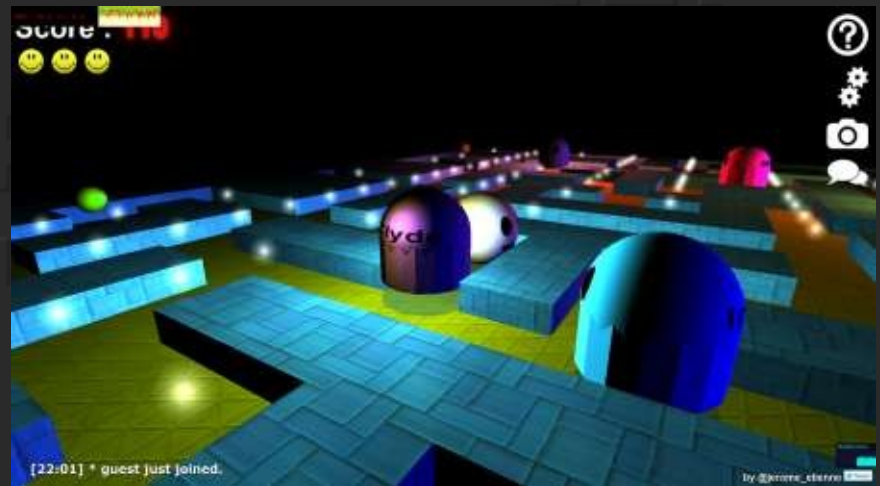
Canvas

- A 32 bit per pixel bitmap and a set of primitives to paint on it
- Feels rather old-school - 0xA0000 anyone?
- Hardware accelerated on some devices / browsers
- Performance varies wildly, but generally getting better
- Nice support for sprites, shapes and fonts



WebGL

- A variant of OpenGL ES 2.0 for Javascript
- Surprise, unsupported on Internet Explorer, WinXP, or iOS Safari
- Programmable shaders through GLSL
- Fantastic performance - necessarily hardware accelerated
 - But keep dynamic geometry and API calls in check



Audio

- The <audio> tag is an absolute disaster for gaming
 - Issues with latency, limits, looping, cloning
 - iOS Safari limits audio usage to user input events
 - iOS Safari only plays back one sound at a time
 - Some implementations unfriendly to some web servers
- Formats: .mp3 vs .ogg (vs .wav?)
- Chrome's Web Audio is the only sane solution IMHO
- Mozilla built Audio Data and now MediaStream Processing API
- SoundManager 2 a common library to deal with this mess
 - Uses Flash audio as fallback
- Some people (esp. on mobile) just opt for sound-less games

Input

- Desktop
 - Mouse events: *click*, *mousedown/mouseup/mousemove*
 - Pointer Lock API: *request/exitPointerLock()*, *pointerlockchange* event, *movementX/Y*
 - Gamepad API: *Gamepad* objects, *connected/disconnected* events
 - Keyboard: *keydown/keyup/keypress*
- Mobile
 - Multitouch: *touchstart/end/move* events and *Touch* objects
 - Mobile devices simulate mouse events but with lag
 - Accelerometer & Geolocation for extra fun
- Remember *event.stopPropagation()* and *event.preventDefault()*

Performance

- Huge variations depending on device and browser
 - Canvas on iOS 4 Safari was about 3x slower than iOS 5
- Rule #1: avoid garbage! Heh
 - Temporary objects have to be garbage collected
 - GC runs can cause pauses several 1/10 of a second
 - Reuse objects, use pools, use *scratch* objects
- Use stable objects
 - Identical internal structure or *type*
 - VMs can JIT functions for inferred types
- General purpose and game-specific optimizations apply
 - Graceful degradation and level of detail

Performance

- Help the compiler with your time-critical code
 - Use local scope, don't use 'with' or deep nesting
 - Move closures and constant expressions out of loops
 - Cache and memoize as much as you can
 - Have good data structures and algorithms before hacking
- Consider Web Workers and TypedArrays
- Learn the tools available for profiling and debugging
- Many surprises await
 - Infamous 'translate-Z-by-zero' to enable hw acceleration
 - Comments can prevent inlining

Mobile wrappers

- Bundle html, Javascript & assets into a native app
- WebView approach
 - Use a native UI component to render Web content
 - Provide hooks for nonstandard native APIs
 - PhoneGap, Appcelerator
 - Performance may suffer compared to native browser
- Runtime + Javascript VM
 - A game engine that exposes Javascript as scripting language
 - Fantastic performance of game-specific primitives
 - GameClosure, CocoonJS, ImpactJS
 - HTML5 and DOM APIs support may be flaky

Online and server side

- AJAX is fine for sending commands to a server
- WebSockets for realtime two-way communications
- NodeJS as a server platform
 - Google's V8 engine + libuv evented I/O library
 - Fast and efficient way to handle lots of traffic
 - Fantastic module ecosystem with its own package manager
 - socket.io is an excellent library & module
 - Clunky for expensive server-side computations
- Databases: MongoDB and Riak support Javascript for Map/Reduce
- WebRTC: Google's project for realtime video/audio communication

Misc

- Careful with load times and bandwidth
 - Consider Cache Manifest and Local Storage / Temp Filesystem
 - Parallelize & preload assets - but not too much!
- Test, test, test - and know when to just ignore a device
- Your source code is in the hands of the enemy
 - Minify, uglify, consider the Closure compiler
 - Think carefully about credentials and cheat prevention
- Don't forget to monetize: Google Wallet, Facebook Credits, etc
- Fantastic prototyping experience
 - Lots of core tech done for you
 - Available everywhere, even mobile

Misc

- Some HTML5 engines
 - Construct 2 - visual editor that outputs HTML5
 - Mandreel - converts C++/C# to WebGL/Flash. Halfbrick-owned
 - ImpactJS - Both a general HTML5 engine and a mobile wrapper
 - Haxe - Actionscript-like, outputs Javascript among others
- Some interesting games to look at
 - BrowserQuest - opensource browser MMO by Mozilla
 - Angry Birds for Chrome - Angry Birds on the Chrome Web Store!
 - Cut The Rope - famous iOS hit ported as a marketing bit for IE9
 - Bejeweled - Granddaddy of Match-3 games
 - The Convergence, Strike Fortress, Agent P, Onslaught Arena

A random act of link spam

- <http://www.html5rocks.com/>
- <http://sixrevisions.com/web-development/html5-iphone-app/>
- <http://blog.nihilogic.dk/2009/02/html5-canvas-cheat-sheet.html>
- <http://creativejs.com/>
- <http://www.lostdecadegames.com/lostcast/>
- <http://learningwebgl.com/blog/>
- <http://code.zynga.com/>
- <http://www.wooga.com/2012/06/woogas-html5-adventure/>
- <http://floitsch.blogspot.de/2012/03/optimizing-for-v8-introduction.html>
- <http://www.html5gamedevelopment.org/>
- <http://browserquest.mozilla.org/>

Thank you!

Any questions?

@TheJare

<http://www.iguanademos.com/Jare/>