IIC3585
Diseño Avanzado de Aplicaciones Web

Semestre 1-2019

Prof. Jaime Navón

Este Curso

- JavaScript, el lenguaje de la Web
  - ES6, ES7 y ES8
  - Typescript
  - WebAssembly
- Nuevos paradigmas
  - Programación funcional
  - Programación reactiva
- Aplicaciones Web Progresivas
  - Por qué son importantes
  - Componentes fundamentales
  - Service Workers
- Frameworks JavaScript
  - ¿Los necesitamos?
  - MV* vs Componentes
  - Los populares (React, Vue, Angular)
- Componentes Web
  - estándares
  - polymer
  - comparación con React
- Optimización para la Web Móvil
  - AMP
  - FIA

Metodología

- Profesor como guía y motivador
- Alumnos estudian temas, hacen desarrollos y ejemplos y los presentan a sus compañeros
- Aprendizaje a partir de trabajo personal y de lo que exponen sus propios compañeros
- No hay pruebas ni examen final
- Notas de presentaciones en clase
  - profundidad de lo aprendido
  - capacidad de traspasarlo a sus compañeros
- Hay un paper final

Evolución de la Web

- Páginas Web
- Sitios Web
- Aplicación Web simple (interfaz web)
- Web 2.0
  - read/write web
  - inteligencia colectiva
  - ajax
  - aplicaciones internet enriquecidas (RIA)
- Web Programable (APIs)
- JavaScript everywhere !
- Web 3.0 (Web Semántica)

Evolución tecnológica

- Establecimiento de estándares: HTTP, HTML, XML, XHTML
- CGI/Perl
- Scripting en Servidor
  - ASP
  - PHP
  - JSP
- Preocupación por Arquitectura de la Aplicación (MVC)
- Frameworks MVC
  - Struts, JSF, Spring, ASP.Net, Zend, Symfony
- Cliente Enriquecido
  - JavaScript, HTML5, CSS3
  - Plugins (Adobe Flash, MS Silverlight)
- Aplicaciones de una página, aplicaciones para móviles
  - Frameworks JavaScript
  - MVC
  - Componentes
  - PWA
- JavaScript en el servidor
- Aplicaciones Isomórficas

JavaScript, una tecnología crucial en la web moderna

La acción migra al cliente

- necesidad de entregar una mejor experiencia de usuario
- necesidad de reemplazar a las aplicaciones nativas
- aparición y rápida difusión de la Web Móvil
- mejoramiento dramático de la implementación de JavaScript en los navegadores

JavaScript en el Cliente

- gran cantidad de código hace necesario mejorar en dos aspectos
  - performance - motor V8
  - manejo de la complejidad
    - librerías (JQuery)
    - frameworks MVC (backbone, ember, angular, react, ...)

Frameworks Javascript

- Permiten separar las componentes modelo-vista-controlador que ahora son cumplidas por el código del cliente
- Fuerza arquitectura Sólida (MVP, MVVM, Flux)
- Implementación de Componentes

Muchos Frameworks

- Backbone.js
- **Ember.js**
- **Angular.js**
- Batman.js
- Knockback.js
- Agility.js
- **React.js**
- **Meteor.js**
- **Aurelia.js**
- **Vue.js**

JavaScript Engines

- Firefox - SpiderMonkey
- Safari - Nitro (JavaScript Core, SquirrelFish)
- Opera - Carakan hasta Opera 15 (v8)
- Chrome, Node.js - V8
- MS Explorer MS Edge - Chakra

https://en.wikipedia.org/wiki/Comparison_of_JavaScript_engines

JavaScript en el Servidor

- Node.js
- Ambiente de programación JavaScript
- V8 Javascript Engine
- Event driven I/O
- non-blocking
- soporta formato standard de módulos JS
- soporta C/C++ add-ons

Non-Blocking I/O

- Servidor pasa mucho tiempo en espera de operaciones I/O
- Todas las operaciones de I/O en Node son asincrónicas
- Event-based en lugar de Thread-based
  - programador crea el thread, ejecuta y elimina el thread
  - servidor crea un nuevo thread por request

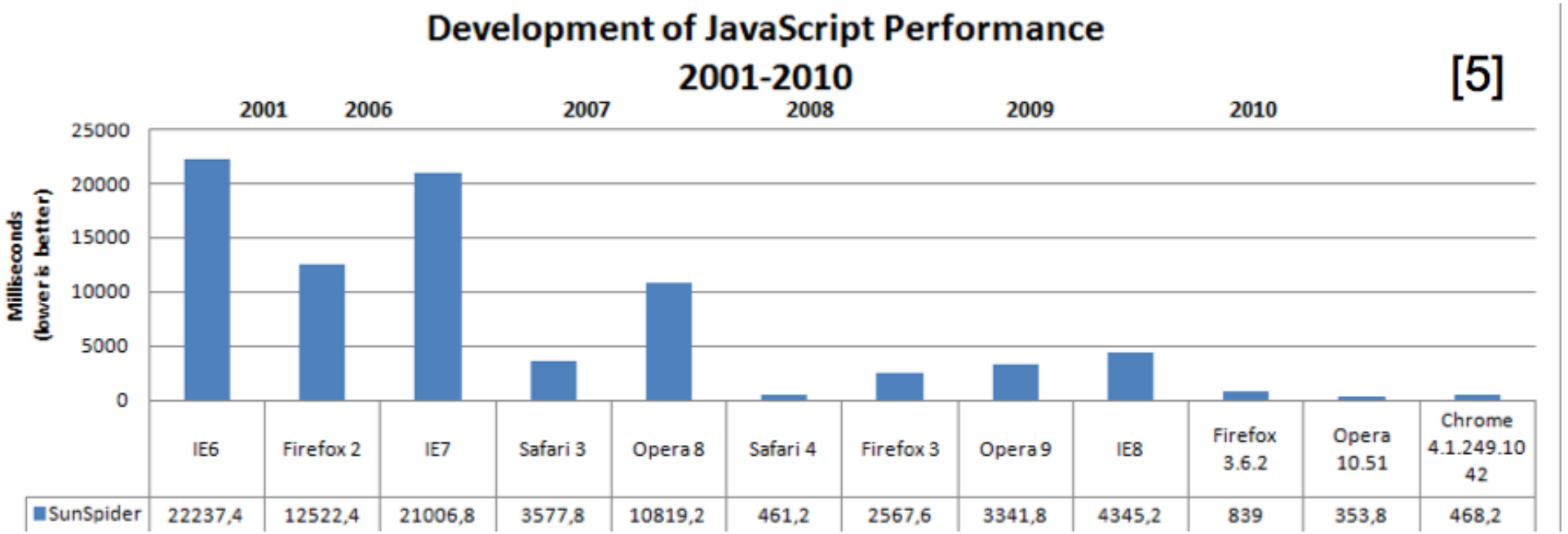
Javascript, la nueva máquina virtual

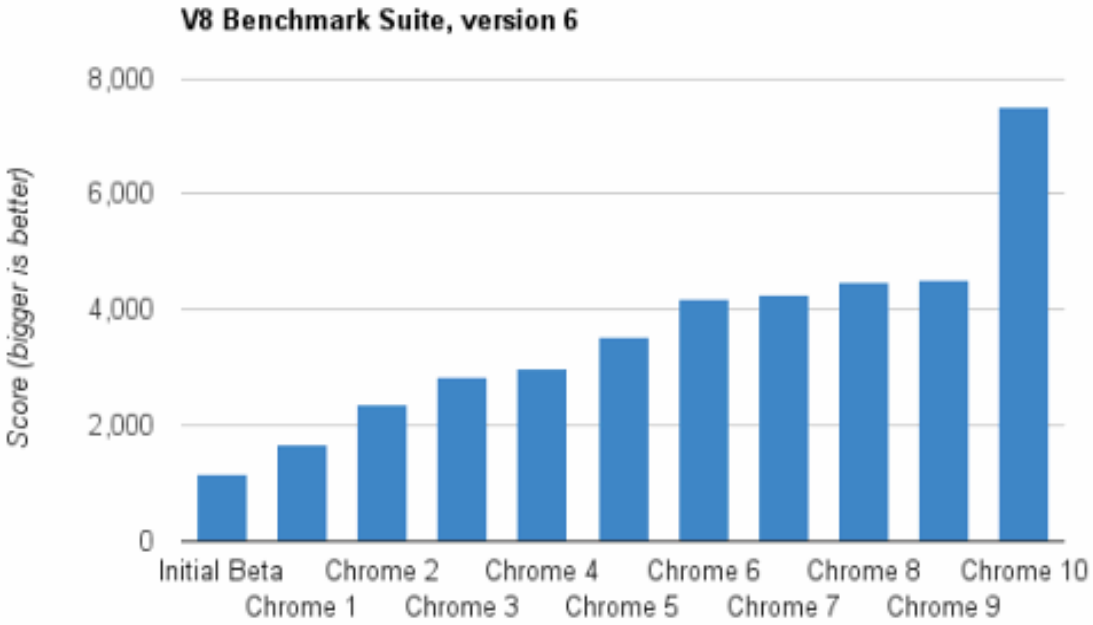
- Pasa a ser la pieza mas importante de una aplicación Web
- Mejoramiento dramático del desempeño del intérprete de JavaScript (hasta 100 veces)
- rapidez de aplicación Web depende principalmente de el desempeño del motor de JavaScript
- JQuery se establece como la librería mas utilizada
- Frameworks JavaScript
- Node
- Herramientas

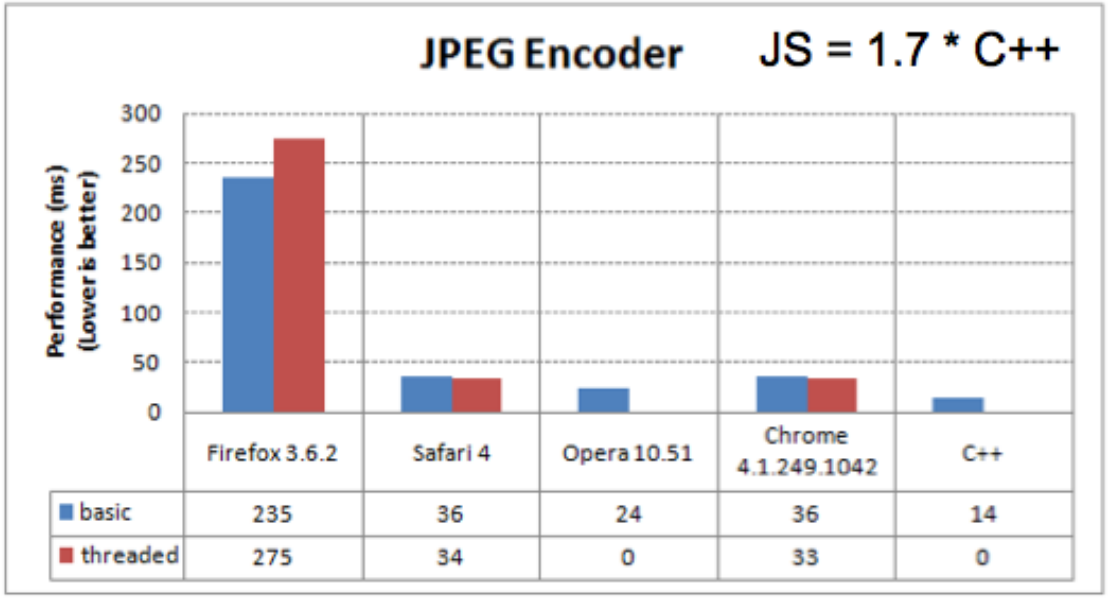
Es realmente rápido

- Apache + PHP
  - 430 req/s @ 60ms promedio
  - 99% @ 114 ms
- Node.js
  - 2000 req/s @ 20ms promedio
  - 99% @ 47ms

Mejoramiento dramático




V8 Benchmark Suite, version 6
Score (bigger is better)
8,000
6,000
4,000
2,000
0
Initial Beta
Chrome 1
Chrome 2
Chrome 3
Chrome 4
Chrome 5
Chrome 6
Chrome 7
Chrome 8
Chrome 9
Chrome 10


JPEG Encoder
JS = 1.7 * C++
Performance (ms)
(Lower is better)
300
250
200
150
100
50
0
Firefox 3.6.2
Safari 4
Opera 10.51
Chrome 4.1.249.1042
C++
basic 235 36 24 36 14
threaded 275 34 0 33 0

JavaScript, un lenguaje de programación interesante

Dustin Díaz, JS extraordinaire (Google)

- JavaScript is the King of Programming languages
- JavaScript has survived bad implementations, bad browser support, and has been flogged by extremely intelligent language experts calling it a "toy" language.
- Programmers of other languages often see JavaScript as a plain, buggy, and unintuitive language.
- JavaScript is colourful, flexible, and easy to work with

Agenda

- Revisión de aspectos nuevos de ES6, ES7, ES8
- Programación Funcional con JS
- Programación Reactiva
- Strong typing (Typescript)
- WebAssembly

Lectura próxima clase

JavaScript brief history and ECMAScript(ES6,ES7,ES8) features

https://medium.com/@madasamy/javascript-brief-history-and-ecmascript-es6-es7-es8-features-673973394df4

The Complete ECMAScript 2015-2017 Guide

https://flaviocopes.com/ecmascript/