

Javascript深入浅出

Yanju Chen

Javascript印象

- The World' s Most Misunderstood Programming Language
 - The Name
 - Mispositioning
 - Design Errors
 - Bad Implementations
 - Substandard Standard
 - Javascript is a Functional Language.

Javascript : 玩具 ?

- A Real Language
- Small, but Sophisticated
- Not A Subset of Java

Javascript诞生记

- Netscape Navigator 0.9



Javascript诞生记

- Oak语言 – Java语言
- Java Applets



Javascript诞生记

- Brendan Eich
- 函数式编程
- Scheme
- 与Java “足够相似”



Javascript诞生记

- 10天设计一门语言
- 思路+特性：
 - 语法：C语言
 - 数据类型&内存管理：Java语言
 - 函数：Scheme语言
 - 继承：Self语言

```
String.prototype.trim =  
function ()  
{  
  return this  
    .replace (/^s+/, "")  
    .replace (/s+$/, "");  
}
```

.js

语言特性

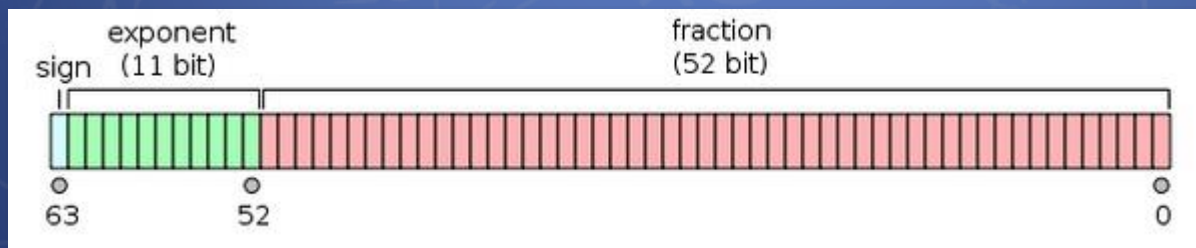
- Small but Sophisticated
 - Load & Go Delivery
 - Loose Typing
 - Objects as General Containers
 - Prototypal Inheritance
 - Lambda
 - Linkage through Global Variables

基本数据类型

- Numbers
- Strings
- Booleans
- Objects
- null
- undefined

Number类型

- IEEE-754
 - double-precision 64-bit format IEEE 754 values
 - ECMAScript : 都是Double问题
 - $0.1 + 0.2 = ?$



$$(-1)^{\text{sign}} \times 2^{\text{exponent} - 0x3ff} \times 1.\text{mantissa}$$

NaN类型

- 告诉你那货不是一个数字.....
- 来源：未定义或非法的操作
- NaN不与任何值相等或相同
 - 就算自己也不行！
 - `NaN===NaN - false`

String类型

- 编码：UCS-2，有点像UTF-16
- 没有单个字符
- 不可修改！！
 - 重新生成效率
- Boolean类型
 - true
 - false

null & undefined

- null : 没定义过
- undefined : 定义过

类型转换

- Number() String() Boolean()
- parseInt() toString()
 - parseInt("08") === 0 第一个非数字字符
 - parseInt("08" ,10) === 8
- +
 - + " 42" = 42
 - '\$' + 3 + 4 = ? 3 + 4 + '\$' = ?
- !!
- % 余数，不是取模

Falsy Values

- false
- null
- undefined
- "" (empty string)
- 0
- NaN

- All other values are truthy.

Evils of Type Coercion

- "" == '0'
- 0 == ""
- 0 == '0'
- false == 'false'
- false == '0'
- false == undefined
- false == null
- null == undefined

Evils of Type Coercion

- `"" == '0'` //false
- `0 == ""` //true
- `0 == '0'` //true
- `false == 'false'` //false
- `false == '0'` //true
- `false == undefined` //false
- `false == null` //false
- `null == undefined` //true

关于变量

- 可以以_和\$开始
- Initial_ should be reserved for implementations.
- \$ should be reserved for machines.

作用域

- {}没有作用域！
- 只有函数有作用域
- Javascript里面函数是以什么形式存在？
 - 先看看Object类型

Object类型

- 键值对 (key-value) , 哈希表存储

- var a={ 'a1' :1, 'a2' : 2' }

- 嵌套

- var a={ 'a1' :1, { 'a11' :11, 'a12' :2}}

- 数组

- 是一种Object

- 特殊表示

- var a=[1,2,3,4];

- var a=['a' , 'b' , 'c'];

Function函数

- First-class Object

- 一种特殊的对象，当成Object的超集
- 有自己的过程方法，自己的成员
- 不需要函数指针
- 动态产生函数
- Inner Function

- 作用域链

- 闭包

Javascript中的闭包

- The scope that an inner function enjoys continues even after the parent functions have returned
- 闭包就是一种内存泄露
- 不释放局部变量
- Javascript中的闭包跟踪
 - 浏览器闭包性能优化

this & arguments

- this
 - 调用者, caller
 - 可以动态改变
- arguments
 - 参数列表
- typeof函数
 - 基本类型测试

命名空间(namespace)

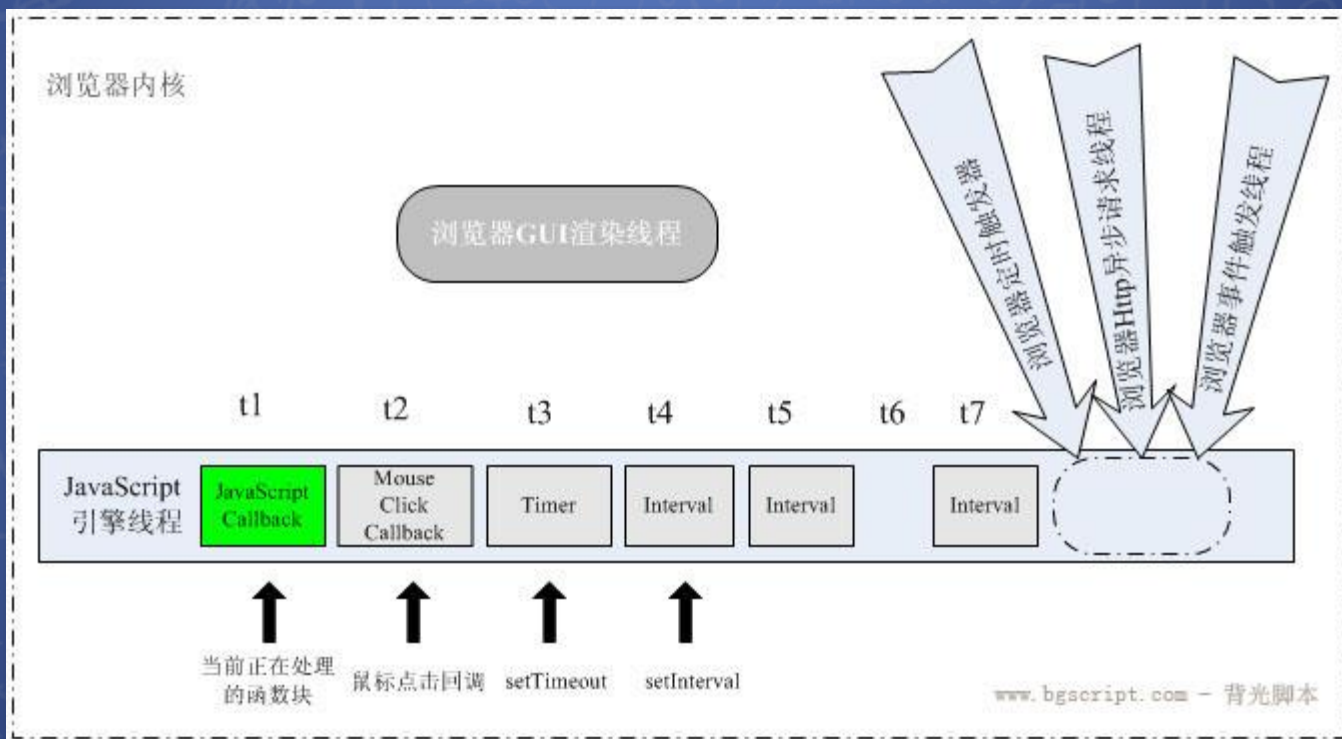
- 每个对象都是独立的命名空间
- 不要滥用全局变量
- 私有变量的实现
 - (function(){}())

关于语言的疑问：

- 面向对象(OO)？
 - 初衷：Prototype继承模型
 - 实际：构造函数方式、原型方式、混合的构造函数/原型方式、动态原型方式
- 多线程(Multi-Threads)？
 - js引擎多数是单线程的，尤其是浏览器中的
 - 基于事件驱动模的模型
 - 伪多线程库Concurrent.Thread



js：事件驱动模型



与服务器交互

- iframe
- img
- Ajax
- Jsonp

- 重要问题：跨域，浏览器同源策略

Node.js : 异质化扩展尝试

- 单线程event loop
- 非阻塞IO
- V8虚拟机
- 事件驱动

- 大量长连接并发问题
 - 快速返回每个请求

Javascript设计缺陷

- 可能成因：
 - 设计阶段过于仓促
 - 没有先例
 - 过早标准化



Javascript设计缺陷

- 不适合开发大型程序
- 非常小的标准库
- null和undefined
- 全局变量难以控制
- 自动插入行尾分号
- 加号运算符
- NaN
- 数组和对象的区分
- ==和===
- 基本类型的包装对象



"foo"	String	string
new String("foo")	String	object
1.2	Number	number
new Number(1.2)	Number	object
true	Boolean	boolean
new Boolean(true)	Boolean	object
new Date()	Date	object
new Error()	Error	object
[1,2,3]	Array	object
new Array(1, 2, 3)	Array	object
new Function("")	Function	function
/abc/g	RegExp	object (function in Nitro/
new RegExp("meow")	RegExp	object (function in Nitro/

Javascript诞生记

- 与其说我爱Javascript，不如说我恨它。它是C语言和Self语言一夜情的产物。
- 十八世纪英国文学家约翰逊博士说得好：'它的优秀之处并非原创，它的原创之处并不优秀。' (the part that is good is not original, and the part that is original is not good.)

参考资料

- [javascript特性杂谈程序即人生 | 程序即人生](#)
- [JavaScript性能陷阱 | 未完待续](#)
- [Node入门](#)
- [Javascript诞生记 - 阮一峰的网络日志](#)
- [Javascript的10个设计缺陷 - 阮一峰的网络日志](#)
- [一段有趣的Javascript代码及分析 - ITeye技术网站](#)
- [\[译\]ES6:JavaScript中将会有的几个新东西 - 紫云飞 - 博客园](#)
- [学习Javascript闭包 \(Closure \) - 阮一峰的网络日志](#)
- [常见语言的REPL\(Read-eval-print-loop\)程序即人生 | 程序即人生](#)
- [Javascript闭包——懂不懂由你，反正我是懂了_知识库_博客园](#)
- [The JavaScript Programming Language](#)
- [重新介绍 JavaScript | MDN](#)
- [对象的消息模型 | 酷壳 - CoolShell.cn](#)
- [Javascript 面向对象编程 | 酷壳 - CoolShell.cn](#)
- [js面向对象-动态原型方法 - prothiphoebe - ITeye技术网站](#)
- [Javascript继承机制的设计思想 - 阮一峰的网络日志](#)