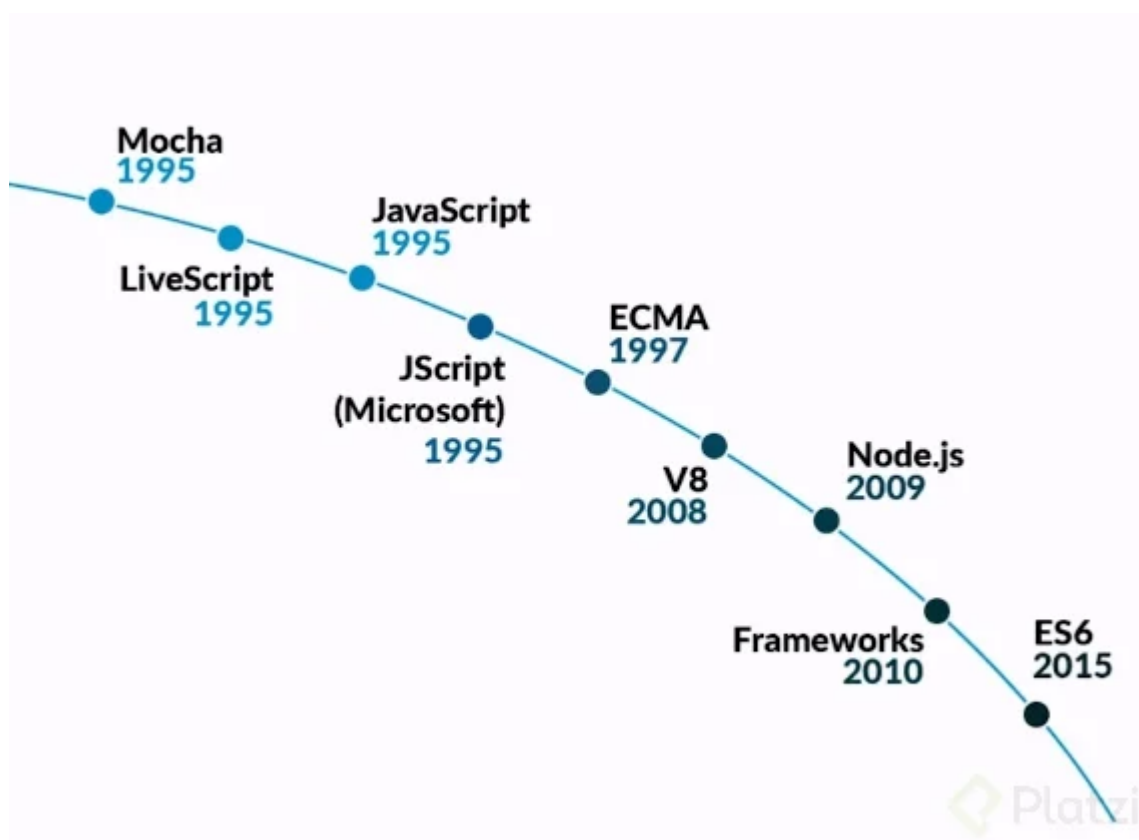


Historia de JavaScript

Con el nacimiento de las tecnologías web (HTTP, HTML y navegadores primitivos), las páginas web eran muy básicas y estáticas: **solo se podía consumir texto e imágenes**.

De hecho, los usuarios del navegador [NetScape](#) solicitaban mejoras y actualizaciones del *software* por la necesidad de evolucionar las páginas web.

JavaScript fue creado por [Brendan Eich](#), inspirado por otros lenguajes como Java, Scheme y Self. El propósito de JavaScript fue dar solución a la problemática de las páginas estáticas.



Origen de JavaScript

El primer lenguaje generado fue *Mocha*, construido por Brendan Eich en 1995. Este **permitía una pequeña interactividad en las páginas** para el navegador *Netscape*. Sin embargo, debido a que su desarrollo fue muy rápido, tuvo que evolucionar, y ese mismo año se produjo *LiveScript*.

LiveScript* era una mejora de *Mocha que ofrecía algunas características de interactividad más avanzadas. A finales del mismo año, se originó una nueva versión del lenguaje, llamada *JavaScript*.

El nombre de *JavaScript* fue puramente comercial, debido a que el lenguaje más popular de la época era Java. Por lo que querían relacionarlo con este para que creciera su popularidad. Sin embargo, JavaScript y Java son dos lenguajes totalmente diferentes.

JavaScript como estándar de la web

A medida que *JavaScript* obtuvo más popularidad, surgieron variaciones de este lenguaje. En 1995, **la empresa Microsoft decidió hacer su variación del lenguaje** para utilizarlo en su navegador Internet Explorer.

Debido a que surgían nuevas versiones del lenguaje, y para evitar que cada navegador tuviera una versión diferente, la organización [ECMA International](#) (que en ese momento surgió como un comité) **decidió que los navegadores emplearían un estándar del lenguaje JavaScript.**

Esta organización se encarga de añadir nuevas características, el lenguaje que permitan la evolución de JavaScript.

Motor Engine V8

En el año 2008, surgió el motor de *JavaScript* creado por Google. Este corre en el navegador, con el nombre [V8 JavaScript Engine](#), y fue un desarrollo importante en la historia de JavaScript, ya que resolvió la problemática de los tiempos de interpretación.

La razón detrás de esta mejora tan grande radica principalmente en la combinación de intérprete y compilador. Hoy en día, los cuatro motores de búsqueda usan esta técnica: el intérprete ejecuta el código fuente casi de inmediato y el compilador traduce el código fuente en código máquina para que el sistema del usuario se ejecute de manera directa.

Node.js, JavaScript del lado del servidor

[Node.js](#) es la forma como podemos utilizar *JavaScript* de parte del servidor o en el backend, esto debido a que utiliza el motor Engine de JavaScript V8.

Mejoras a JavaScript

Un **framework** es un conjunto de herramientas y funcionalidades para **resolver un problema de software**. Para *JavaScript*, los *frameworks* permiten construir páginas web de manera rápida y optimizada.

En el 2010 empiezan a surgir los primeros frameworks de JavaScript.

La organización encargada de los estándares, *ECMA International*, decidió modernizar el lenguaje con mejoras que permitieran trabajar más rápido y óptimo con JavaScript.

En el 2015 se publicó ECMAScript 6 o ECMAScript 2015 que **recopiló nuevas características del lenguaje** y se publicó como una nueva versión de *JavaScript*.

Cada año, una nueva versión de *JavaScript* es publicada con el nombre de ECMAScript 7, 8, 9, y así sucesivamente. Esto significa que existirán nuevas características como estándar en el lenguaje que cada navegador implementará.