



AJAX

Dinamizando la web



“

¿Qué es Ajax?

”

Asynchronous JavaScript And XML

AJAX permite actualizar las páginas web de forma asíncrona intercambiando datos con un servidor web tras bambalinas. Esto significa que es posible actualizar partes de una página web, sin recargar la página entera.



¿Qué lo compone?

Un objeto XMLHttpRequest incorporado en el navegador (para solicitar datos desde un servidor web).

JavaScript y HTML DOM (para mostrar o utilizar los datos)

¿Cómo funciona Ajax?

Browser

Ocurre un evento:

1. Se crea un objeto XMLHttpRequest
2. Se envía el HttpRequest

Ocurre un evento:

1. Se crea un objeto XMLHttpRequest
2. Se envía el HttpRequest

Server

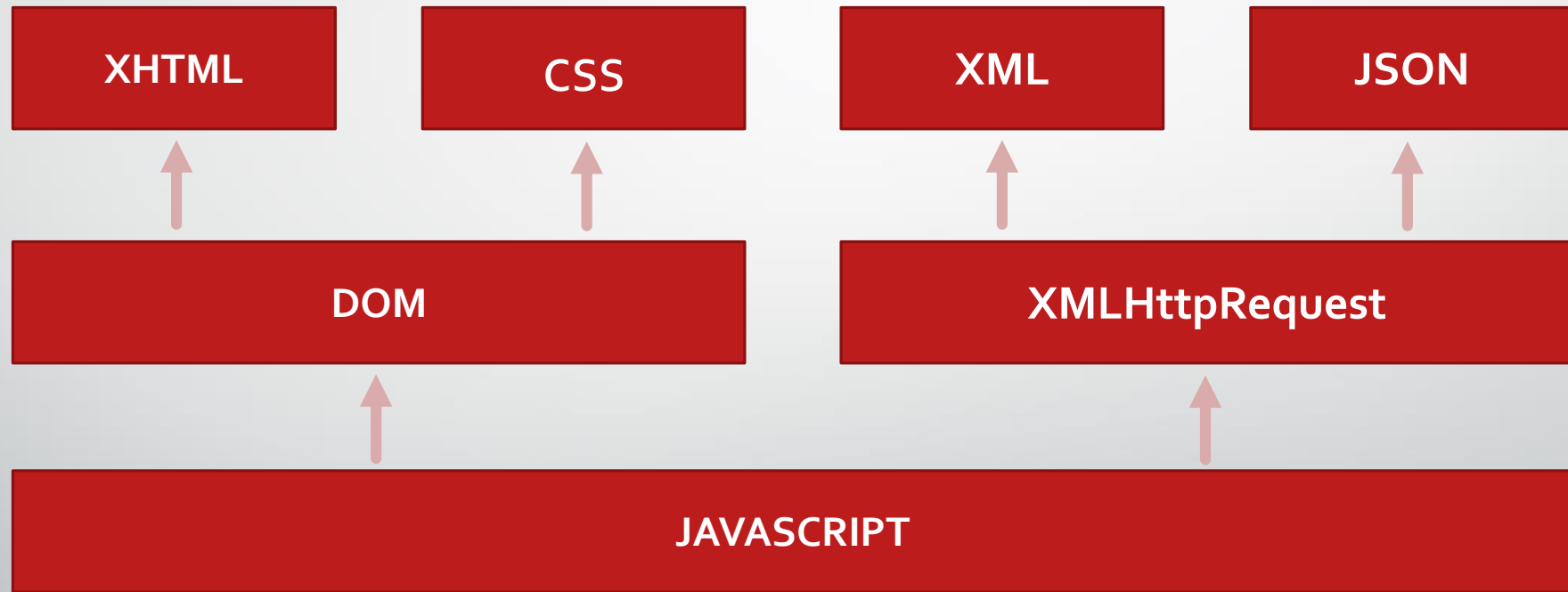
1. Se procesa el HttpRequest
2. Crea una respuesta y envía los datos al browser



Paso a paso

1. Se produce un evento en una página web (la página se carga, se hace clic en un botón)
2. Un objeto XMLHttpRequest es creado por JavaScript
3. El objeto XMLHttpRequest envía una solicitud a un servidor web
4. El servidor procesa la solicitud
5. El servidor envía una respuesta a la página web
6. La respuesta es leída por JavaScript
7. La acción apropiada es realizada por JavaScript

Interacción de tecnologías





“

JSON (JavaScript Object Notation - Notación de Objetos de JavaScript) es un formato ligero de intercambio de datos.

”

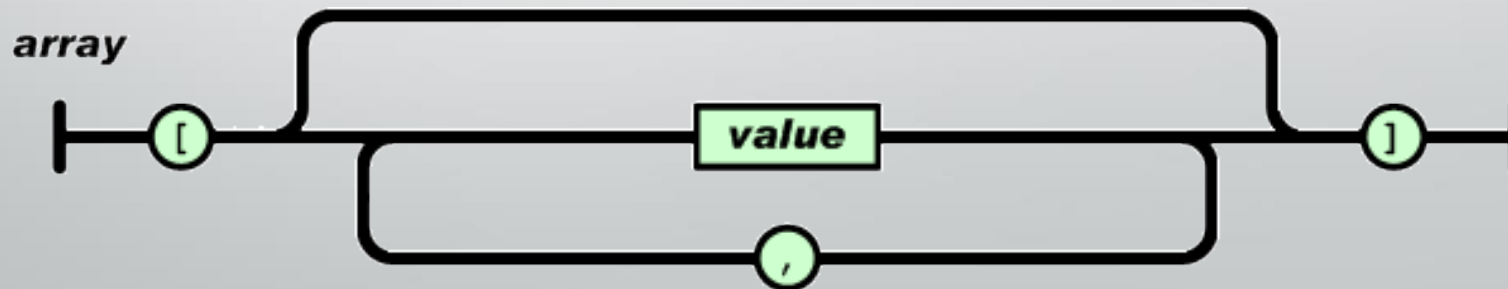
JSON es un formato de texto que es completamente independiente del lenguaje, JavaScript es uno de los lenguajes que lo soporta.


¿Cómo se forma un JSON?

Un objeto es un conjunto desordenado de pares nombre/valor.

Un objeto comienza con { (llave de apertura) y termine con } (llave de cierre).

Cada nombre es seguido por : (dos puntos) y los pares nombre/valor están separados por , (coma).





Ejemplo

{ nombre: Belen, apellido: Alegre, nacionalidad: Argentina};

¿GET o POST?

Generalmente GET suele ser más rápido, pero existen casos en los cuales debe utilizarse POST.

1. Envío de una gran cantidad de datos al servidor (POST no tiene limitaciones de tamaño).
2. Enviando un input de usuario (que puede contener caracteres desconocidos), POST es más robusto y seguro que GET .