



UD1: Arquitecturas y lenguajes de programación en clientes Web

Desarrollo Web en Entorno Cliente

Índice

- [Arquitectura de aplicaciones Web](#)
- [Introducción a Javascript](#)
- [Componentes de un navegador Web](#)



Arquitectura de aplicaciones Web

[Indice](#)

UD1: Arquitecturas y lenguajes de programación en entornos Web

Nacimiento de la Web

- Concebida y creada por Tim Berners-Lee en 1989
 - Especialista del CERN
 - Presenta el World Wide Web (www)
- Primer documento formal: 1991.
 - Internet existía desde los 60.
- Web: 3 partes:
 - **HTML**: Lenguaje de marcado de **hipertexto**.
 - **URL**: Modo de acceso.
 - **HTTP: Protocolo** de comunicación entre cliente Web y servidor Web.
- Protocolos y especificaciones adaptados universalmente



Estandarización en la Web

- Consorcio W3C

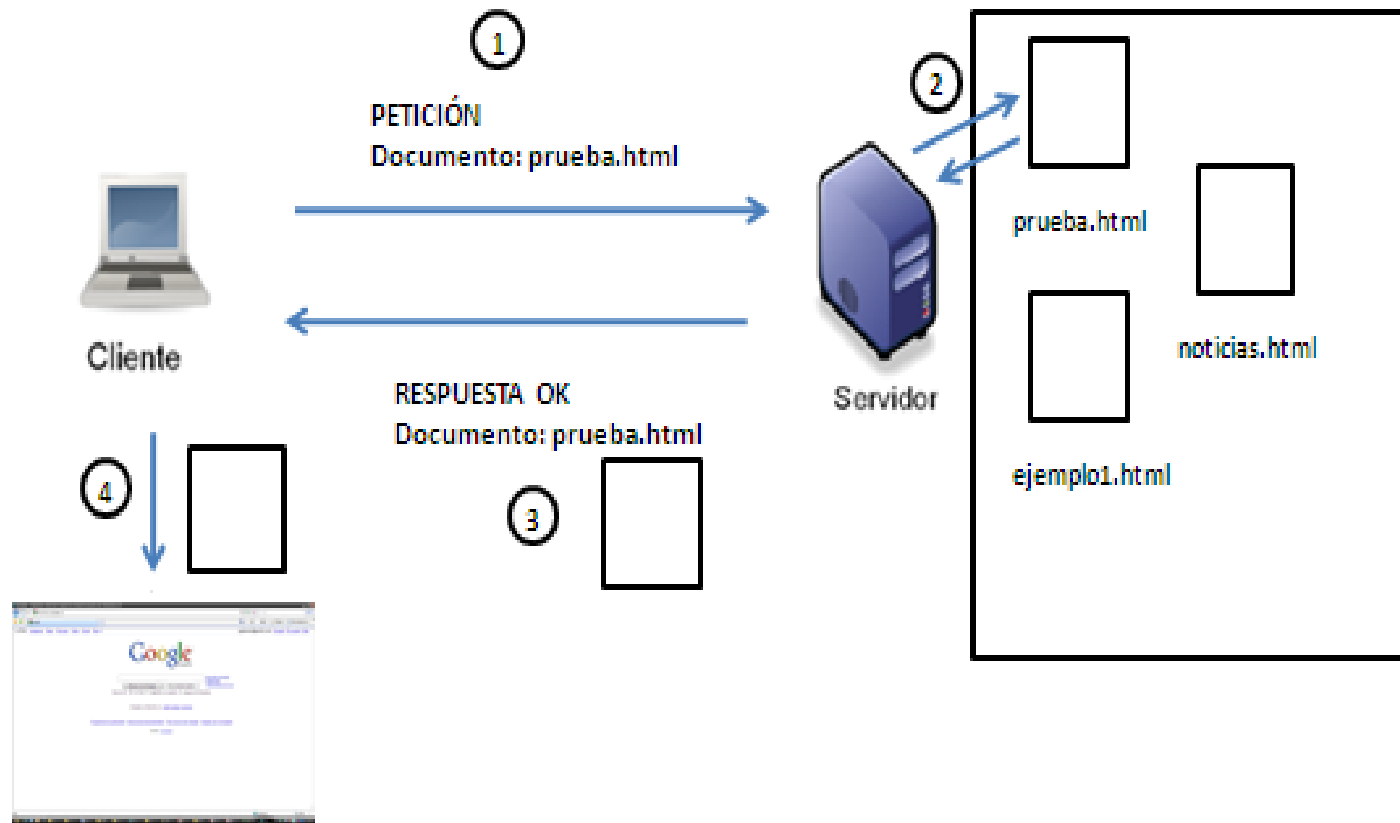
- Organización que se responsabiliza de desarrollar y mantener los estándares Web
- Entre sus [miembros](#) están organismos oficiales, empresas, universidades..



La Web

- Funciona siguiendo el modelo cliente-servidor.
 - Servidor: Entidad que presta el servicio.
 - Se incluye el hardware y software del servidor Web:
 - Scripts y programas CGI
 - Programas API del servidor
 - Servlets
 - Lenguajes PHP, ASP.
 - Cliente. Entidad que recibe el servicio.
 - Hace referencia a los navegadores Web y está soportado por tecnologías HTML, CSS y lenguajes como JavaScript y controles Active X:
 - Plug-ins y programas API del navegador
 - Applets de Java
 - JQuery
- Comprender todos los aspectos del modelo Web es fundamental para ser un buen Desarrollador Web.

Petición de una página Web

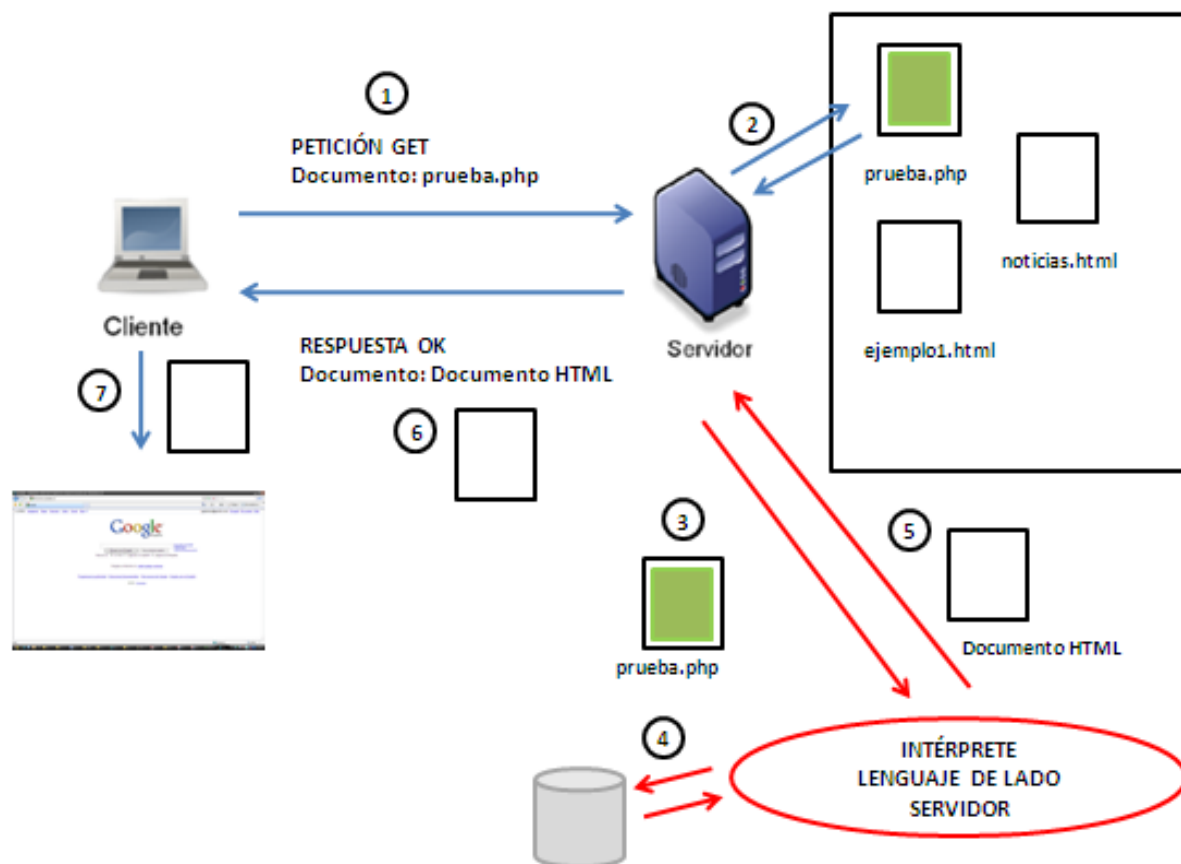


El lado servidor

- Servidor Web (hardware y software)
- Tecnologías y lenguajes de lado servidor
 - **PHP**, ASP, JSP..
 - Java/CGI/Perl
 - Bases de datos



Lenguajes de lado servidor

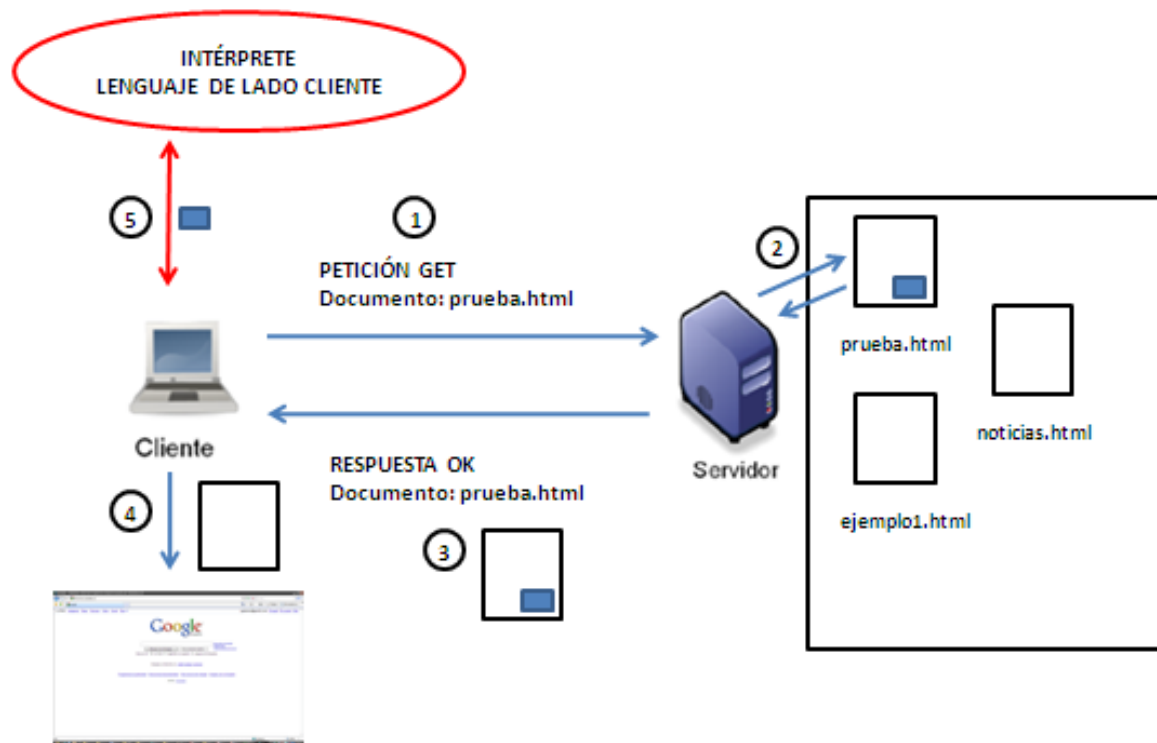


Tecnologías Web de lado cliente

- Navegador Web
- Lenguajes entendidos por el navegador:
 - HTML/CSS
 - **Javascript**: De manera nativa por todos los ordenadores.
 - **Flash** (Adobe), **Silverlight** (Microsoft): Plugin adicional.

Tecnologías Web de lado cliente

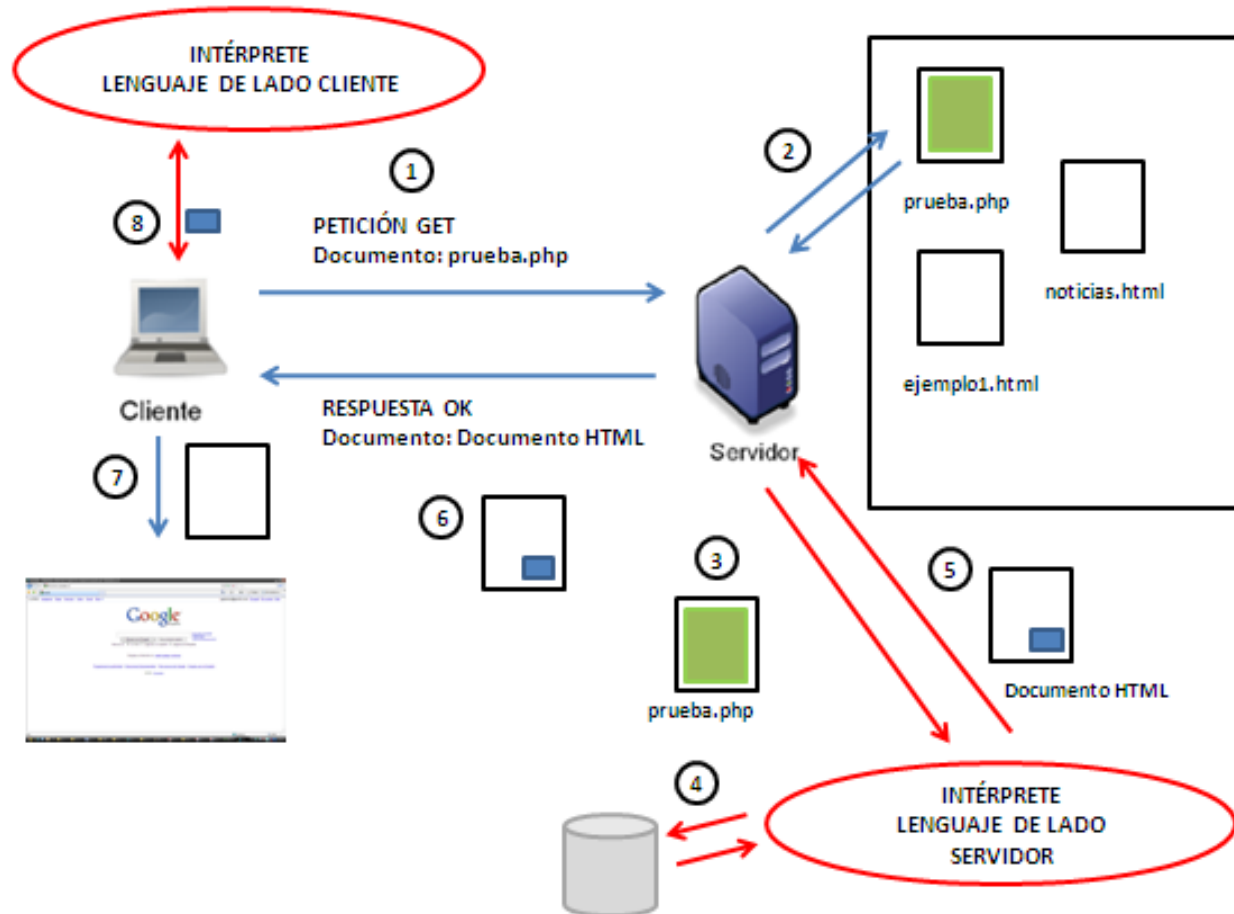
- El código se representa mediante un cuadro azul en la página.



¿Cliente o Servidor?



Cliente y Servidor



- Es una tecnología a medio camino entre cliente y servidor.
- Evita tener que hacer nuevas peticiones de la página completa para cada recurso que se pide.
 - Sólo se pide la parte que ha cambiado.
 - No es necesario refrescar la página completa, permitiendo aplicaciones mucho más complejas.

Y para complicar más el tema...

- Node.js
 - Entorno que permite programar la capa de servidor con código Javascript





Introducción a Javascript

[Indice](#)

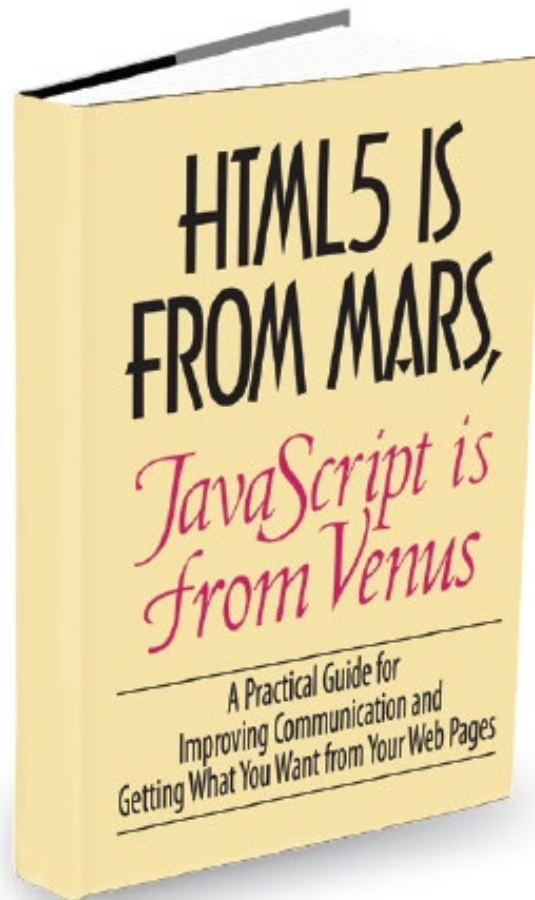
UD1: Arquitecturas y lenguajes de programación en entornos Web

¿Por qué Javascript?



- HTML5 incorpora nuevas características que ayudan a superar las limitaciones de HTML
 - Gráficos, almacenamiento de datos, etc..
- Gran parte de las funcionalidades de HTML5 se definen como bibliotecas de funciones Javascript.
- Es por ello que Javascript es el lenguaje por defecto para HTML5
 - Van de la mano aunque sean muy distintos.

Es decir...



Javascript: Algo de historia

- Originalmente creado para el navegador NetScape (1995).
 - Nombres: Mocha → LiveScript → JavaScript.
 - Desarrollado en [10 días](#)!
 - Comparte el nombre con Java por razones comerciales
- Poco después Microsoft sacó su propio lenguaje: Jscript
 - Hoy en desuso pero llegó a ser competencia.



ECMAScript

- Javascript es estandarizado por ECMA desde 1997
 - Su versión “**oficial**” se llama ECMAScript.
- Últimas versiones:
 - Versión 6 (ECMAScript 2015).
 - Características de código habituales en otros lenguajes
 - Versión 7 (ECMAScript 2016)
 - Mejoras en Array
 - Versión 8 (ECMAScript 2017)
 - Memoria compartida
 - Funciones asíncronas
 - Versión 9 (ECMAScript 2018)
 - Interacciones asíncronas

Usos principales de Javascript

- Interacción con el usuario.
- Distribución de grupos de datos a lo largo de la interfaz.
- Pre-proceso antes de enviarlo al servidor.
- Modificar estilo y contenido de manera dinámica en función de las acciones del usuario.

Compatibilidad

- Es un estándar desde 1997.
 - Esto implica que es aceptado por todos los navegadores
 - Pero:
 - **Puede haber diferencias en su tratamiento**
 - *En muchas ocasiones la diferencia de tratamiento proviene del HTML (**validar**)*
 - **Puede ser deshabilitado en el navegador**

Principales características

- Lenguaje interpretado.
 - Se ejecuta en el navegador línea a línea.
 - No necesita ser compilado.
- Orientado a objetos
- Tipado débil
 - No tenemos que indicar el tipo de las variables.
- Tipado dinámico.
 - Podemos cambiar el tipo de las variables en tiempo de ejecución.

Seguridad

- JS se ejecuta en **sandbox**.
 - Esto significa que no puede acceder a recursos externos al navegador, como el explorador de archivos, periféricos, etc...
- Política de mismo origen
 - Desde los scripts que provienen de una Web **en principio** no tengo acceso a datos enviados desde otras Webs

Ventajas

- Estándar.
- Rápido (cada vez más).
- Versátil (cada vez más).
- Muy compatible con dispositivos de todo tipo.





Componentes de un navegador Web

[Indice](#)

UD1: Arquitecturas y lenguajes de programación en entornos Web

Estructura de un cliente Web

- Podemos entender que en el lado cliente tenemos 4 capas:

Comportamiento (Javascript)

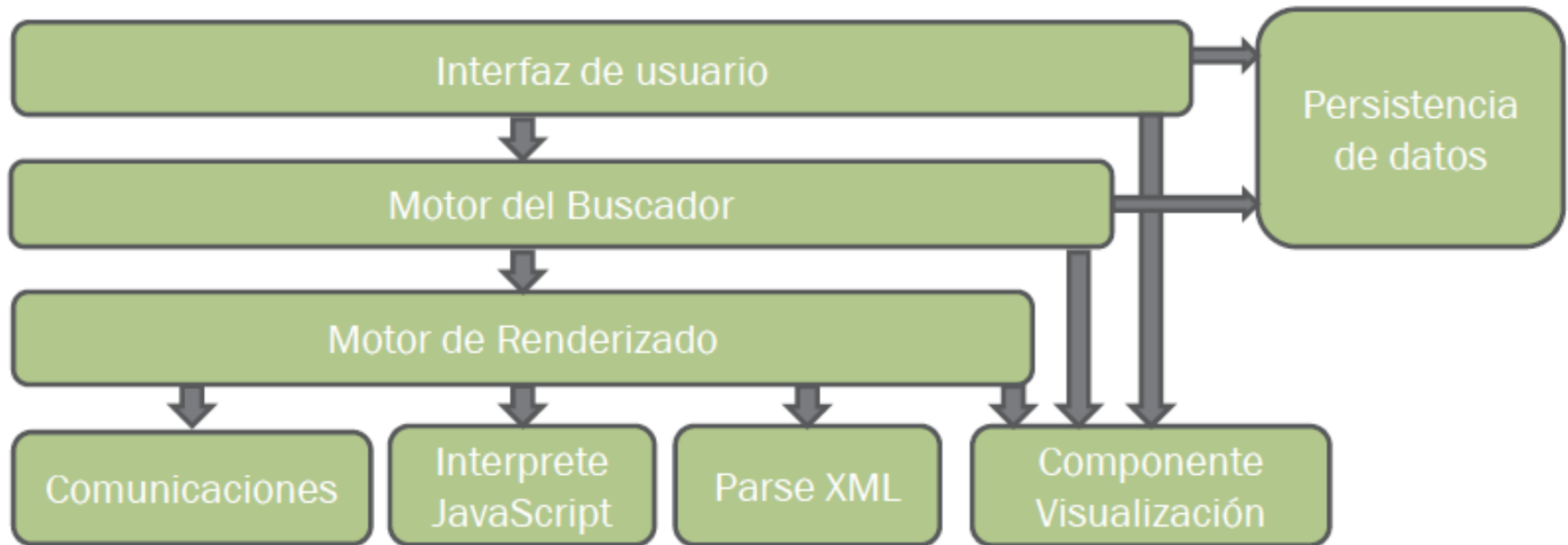
Presentación (CSS)

Estructura

Contenido (texto,
imágenes, MM, etc)

HTML

Arquitectura de un Navegador Web



Interfaz de usuario



- Recoge las acciones de usuario.
 - Barra de direcciones
 - Botones de navegación
 - Preferencias de configuración.
 - ...
- Todo salvo la parte donde se muestra la página

Motor del buscador

- Comunica las acciones desde la interfaz de usuario a la capa de renderizado.



Capa de renderizado (motor)

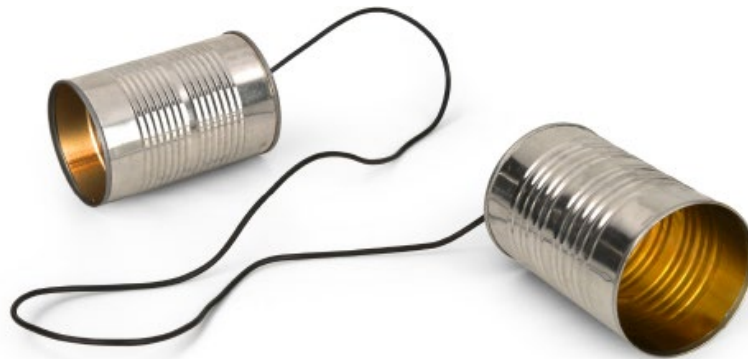


- Produce una representación visual del recurso obtenido.
- Interpreta el código y muestra el sitio Web junto con el contenido (imágenes, audio/vídeo...)

Navegador	Motor
Mozilla	Gecko
Internet Explorer, [Edge]	Trident, [EdgeHTML]
Safari, [Chrome, Opera]	WebKit [Blink]

Capa de comunicaciones

- Protocolo de transferencia de ficheros y documentos (HTTP, FTP)..
- Gestiona la **caché** de elementos accedidos recientemente.



Componente de visualización

- Ofrece primitivas de dibujo y posicionamiento en una ventana y conjunto de fuentes tipográficas.
- Relacionado con las librerías de visualización del S.O.



Intérprete de Javascript

- Analiza y ejecuta el código JS
- Puede ser configurado, e incluso deshabilitado, desde el motor de navegación o el de renderizado.
- Cada navegador tiene su propio intérprete.
 - Ejemplo: Chrome usa V8, también usado en Node.js

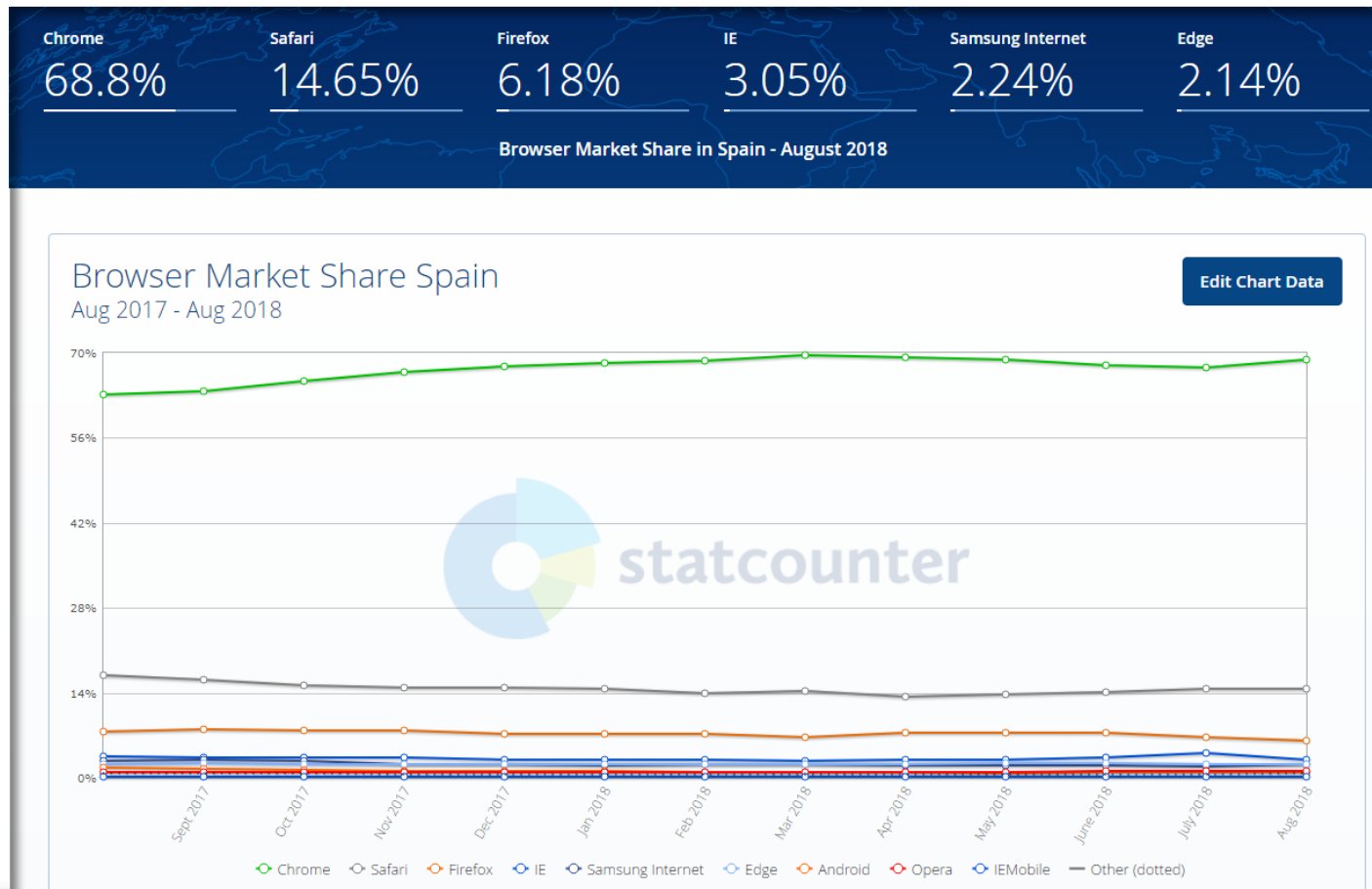
Persistencia de datos

- Almacén de datos para los distintos subsistemas del navegador.
- Existen distintos mecanismos:
 - Cookies
 - localStorage
 - IndexedDB
 - WebSQL
 - FileSystem



¿Qué navegador es el más usado?

- Estadística de uso de navegadores Web (España)



¿Qué navegador es mejor?

- La elección debería hacerse en base a criterios como:
 - Rapidez
 - Rendimiento general
 - Plugins y añadidos
 - Compatibilidad
- Algunas comparativas
 - [The best browser in 2018](#)
 - [Battle of the browsers](#)
 - [Best Browsers os 2018 \(incluye benchmarks\)](#)

