

## TEMA 1

### SELECCIÓN DE ARQUITECTURAS Y HERRAMIENTAS DE PROGRAMACIÓN

#### 1. Tipos de páginas web

Una sencilla clasificación de los tipos de páginas web podría ser esta

- Páginas estáticas
- Páginas dinámicas

##### 1.1. Páginas estáticas

Diremos que una página es estática cuando sus contenidos no pueden ser modificados –ni desde el servidor que la aloja (ordenador remoto) ni tampoco desde el cliente (navegador)– mediante ninguna intervención del usuario ni tampoco a través de ningún programa.

##### 1.2. Páginas dinámicas

Llamaremos dinámicas a las páginas cuyos contenidos sí pueden ser modificados –de forma automática o mediante la intervención de un usuario– bien sea desde el cliente y/o desde el servidor. Para que esas modificaciones puedan producirse es necesario que algo o alguien especifique: qué, cómo, cuándo, dónde y de qué forma deben realizarse, y que exista otro algo o alguien capaz de acceder, interpretar y ejecutar tales instrucciones en el momento preciso. Igual que ocurre en la vida cotidiana, las especificaciones y las instrucciones requieren: un lenguaje para definirlas; un soporte para almacenarlas y un intérprete capaz de ejecutarlas.

#### Scripts

Se llama script a un conjunto de instrucciones escritas en un lenguaje determinado que van incrustadas dentro de una página WEB de modo que su intérprete pueda acceder a ellas en el momento en el que se requiera su ejecución.

Cuando se incrustan scripts en una página WEB empiezan a convivir en un mismo documento informaciones destinadas a distintos intérpretes. Por una parte, el código HTML que ha de ser interpretado por el navegador, y por la otra, los scripts que han de ser ejecutados –dependiendo del lenguaje en el que hayan sido escritos– por su intérprete correspondiente. La manera de diferenciar los contenidos es delimitar los scripts marcando su comienzo con una etiqueta de apertura `<script>` y señalando el final con una etiqueta de cierre `</script>`.

Lo que no está contenido entre esas etiquetas se considerará código HTML.

La posibilidad de insertar en un mismo documento scripts desarrollados en distintos lenguajes obliga a especificar cuál se ha

utilizado en cada caso, para que en el momento en el que vayan a ser ejecutados se invoque el intérprete adecuado.

Para ello, dentro de la propia etiqueta de apertura (<script>) se inserta una referencia al tipo de lenguaje con esta sintaxis: language="nombre"

Por ejemplo:

```
<script language="PHP"> ó <?php>
.....
..... instrucciones ..
.....
</script>
```

indicaría que las instrucciones están escritas con la sintaxis de PHP. Por el contrario, en este otro supuesto:

```
<script language="JavaScript">
.....
..... instrucciones ..
.....
</script>
```

estaríamos señalando que en las instrucciones contenidas en el script se ha utilizado sintaxis de JavaScript.

## 2. Lenguajes

Hay múltiples posibilidades en cuanto a lenguajes de script. Pero antes de hacer mención a algunos de ellos es conveniente hacer una clasificación previa.

Los lenguajes de script pueden clasificarse en dos tipos:

- Del lado del cliente
- Del lado del servidor

### 2.1. Lenguajes del lado del cliente

El navegador es una especie de aplicación capaz de interpretar las órdenes recibidas en forma de código HTML fundamentalmente y convertirlas en las páginas que son el resultado de dicha orden.

Cuando nosotros pinchamos sobre un enlace hipertexto, en realidad lo que pasa es que establecemos una petición de un archivo HTML residente en el servidor (un ordenador que se encuentra continuamente conectado a la red) el cual es enviado e interpretado por nuestro navegador (el cliente).

Sin embargo, si la página que pedimos no es un archivo HTML, el navegador es incapaz de interpretarla y lo único que es capaz de hacer es salvarla en forma de archivo. Es por ello que, si queremos emplear lenguajes accesorios para realizar un sitio web, es

absolutamente necesario que sea el propio servidor quien los ejecute e interprete para luego enviarlos al cliente (navegador) en forma de archivo HTML totalmente legible por él.

De modo que, cuando pinchamos sobre un enlace a una página que contiene un script en un lenguaje comprensible únicamente por el servidor, lo que ocurre en realidad es que dicho script es ejecutado por el servidor y el resultado de esa ejecución da lugar a la generación de un archivo HTML que es enviado al cliente.

Diremos que un lenguaje es del lado del cliente cuando son directamente interpretados por el navegador y no necesitan pretratamiento.

Seguramente te ha ocurrido alguna vez que al intentar acceder a una página web ha aparecido un mensaje diciendo que la correcta visualización de la página requiere un plug-in determinado, y que, a la vez, se te haya ofrecido la posibilidad de descargarlo en ese momento.

Eso ocurre porque cuando el navegador –que en el caso de las páginas web es el cliente– trata de interpretar la página, encuentra incrustado en ella algo (un fichero de sonido, una animación Flash, etcétera) que –de forma muy similar a lo que ocurre con los scripts– requiere un intérprete adecuado del que no dispone en ese momento. Cuando los scripts contenidos en un documento son de este tipo, el servidor lo entrega al cliente sin efectuar ningún tipo de modificación.

Algunos lenguajes del lado cliente son:

- DHTML
- JavaScript
- VBScript

**DHTML** no es exactamente un lenguaje de programación. Se trata más bien de una serie de capacidades que se han ido añadiendo a los navegadores modernos mediante las cuales las páginas pueden contener hojas de estilo y/o organizarse en capas susceptibles de ser redimensionadas, modificadas, desplazadas y/o ocultas. Puede considerarse una combinación de HTML, CSS y JavaScript

**JavaScript** es uno de los lenguajes más populares. Cada navegador incluye su propio intérprete y es frecuente que los resultados de visualización sean algo distintos según el navegador y la versión que se utilice.

**VBScript** es un lenguaje de script derivado de VisualBasic y diseñado específicamente para los navegadores de Microsoft.

**Applets** programados en Java. Son pequeños programas que se transfieren precompilados con las páginas web y que el navegador ejecuta a través de un módulo o extensión concretos.

Una de las principales ventajas de utilizar applets es que son mucho menos dependientes del navegador que los scripts de JavaScript. Como desventajas notables frente al uso de JavaScript, cabe señalar que son más lentas de procesar y que tienen un espacio delimitado en la página donde se ejecutan, es decir, no se mezclan con todos los elementos de la página ni tienen acceso a ellos. Como consecuencia directa, con los applets no podremos realizar directamente acciones tales como abrir ventanas secundarias, controlar marcos, obtener información de formularios, etc.

## 2.2. **Lenguajes del lado del servidor**

Un lenguaje es del lado del servidor cuando la ejecución de sus scripts se efectúa antes de dar respuesta a la petición, de manera que el cliente no recibe el documento original sino el resultante de esa interpretación previa. Cuando se usan estos tipos de lenguaje el cliente recibe un documento en el que cada script contenido en el original habrá sido sustituido por los resultados de su ejecución. La utilización de este tipo de scripts requiere que el intérprete del lenguaje sea accesible desde el propio servidor. ¿Cómo resuelve sus dudas el servidor? Dado que en unos casos el servidor debe entregar el documento original –páginas estáticas o páginas dinámicas en las que se usan lenguajes del lado del cliente– mientras que en otros casos –páginas dinámicas usando lenguajes del lado del servidor– tiene que devolver el resultado de la ejecución de los scripts, es razonable que te preguntes:

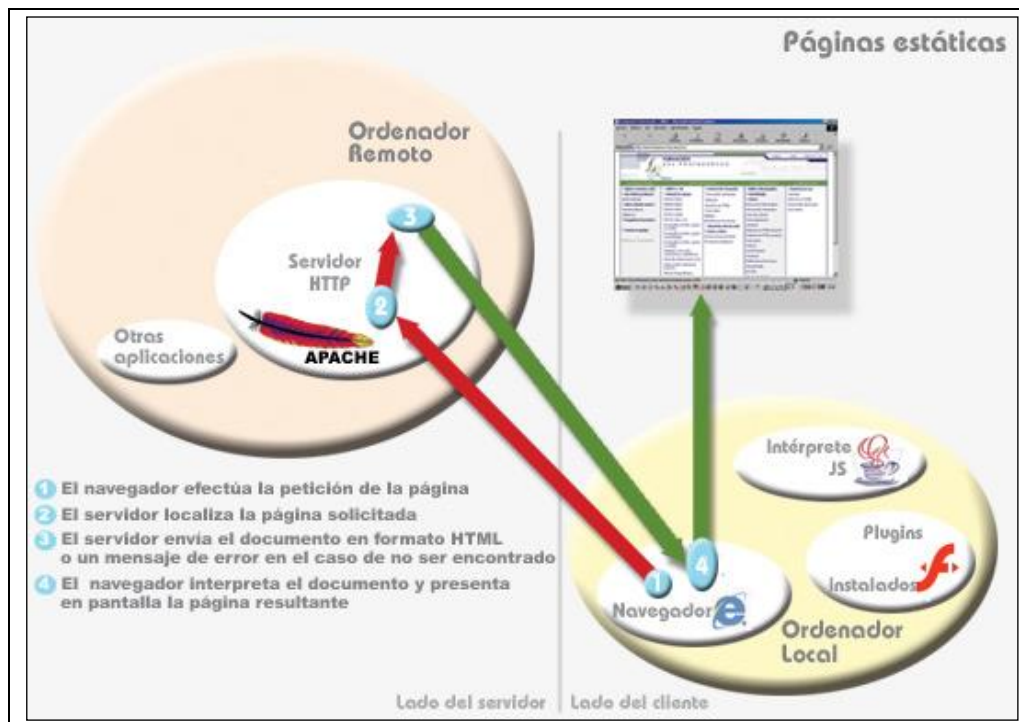
¿cómo sabe el servidor lo que debe hacer en cada caso? La respuesta es simple. Eso hay que decírselo. Y se le dice de una forma bastante simple. Se indica al poner la extensión al documento.

Los scripts son almacenados en el servidor quien los ejecuta y traduce a HTML por lo que permanecen ocultos para el cliente. Este hecho puede resultar a todas luces una forma legítima de proteger el trabajo intelectual realizado.

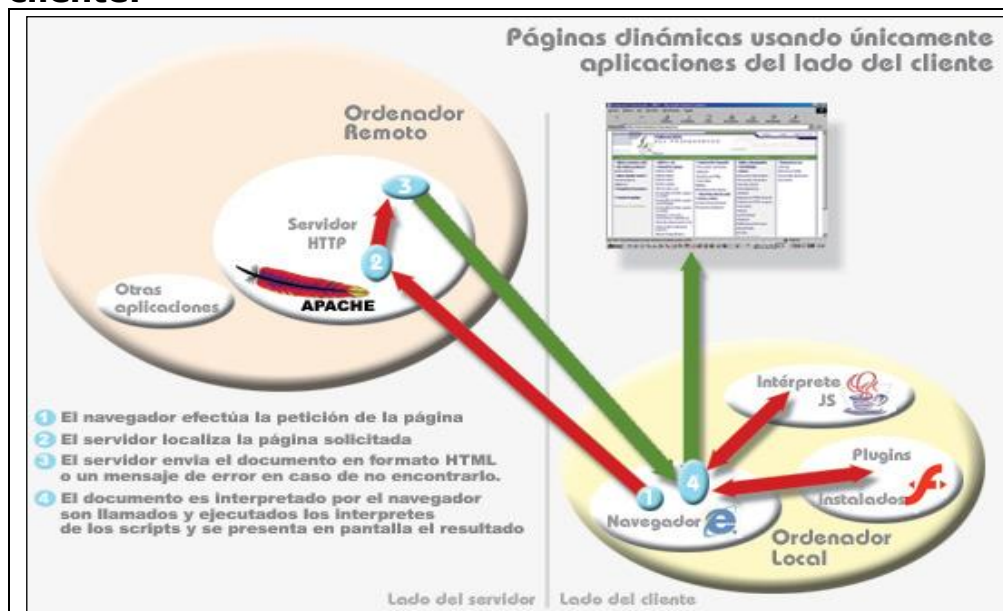
Algunos lenguajes del lado servidor son PHP, ASP, Java, Phyton, etc

### 3. Esquemas de diferentes peticiones de páginas WEB

#### 1) Petición de página sin código script incrustado:

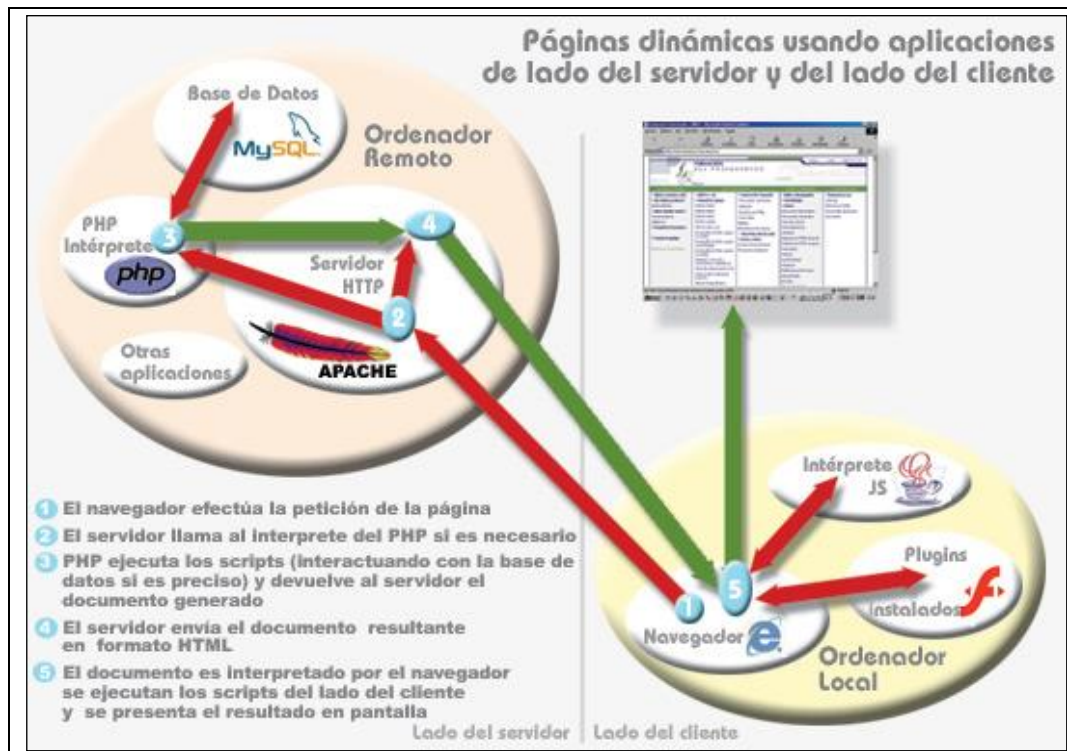


#### 2) Petición de página en la que hay incrustados scripts escritos en lenguaje del lado del cliente:



en este caso se harían llamadas al intérprete de JavaScript –incluido en los navegadores– y/o a eventuales plugins necesarios para interpretar otros tipos de script.

### 3) Convivencia en un mismo documento de varios scripts y varios tipos de lenguaje:



Aquí ya es preciso que, además de un servidor capaz de soportar el protocolo HTTP, esté instalado –del lado del servidor– un intérprete PHP, un servidor de bases de datos MySQL y que, además, estén configurados de modo que puedan interactuar entre ellos.

## 4. Complementos para navegadores

Los complementos para los navegadores añaden funciones que facilitan el desarrollo de las páginas web, como consultar las propiedades CSS que tiene un elemento de la página web, o que permiten la utilización de algún servicio en Internet, como la validación del código HTML o CSS de una página y pueden hacer crecer tu navegador con nuevas características, diccionarios de otros idiomas o cambiar su aspecto visual.

Para ver las extensiones, en Chrome se ha de escribir lo siguiente: chrome://extensions y en Mozilla about:addons

En Internet Explorer: Herramientas+Administrar complementos

A continuación se incluye una lista de complementos para los navegadores [Chrome](#), [Microsoft Internet Explorer](#), [Mozilla Firefox](#) y [Opera](#).

### Todos los navegadores

#### [Complex Table Inspector](#)

Un favelet (pequeño fragmento de código JavaScript que se almacena como "favorito" en el navegador) que permite visualizar la información sobre una tabla que suele estar oculta en los navegadores visuales: summary, headers, axis, scope y abbr.

#### [Favelets For The Validator](#)

Varios favelets del [W3C](#) que permiten validar el código HTML de la página que se está visualizando.

#### [NCAM Accessibility QA Favelet](#)

Un favelet que realiza varios análisis que permiten localizar problemas de accesibilidad en una página web.

### Chrome

#### [Chrome Web Developer Tools](#)

Un clon del complemento [Web Developer](#) para Mozilla Firefox.

#### [Firebug Lite](#)

Permite editar y corregir errores en el HTML, CSS y JavaScript en "vivo" en cualquier página web.

#### [IE Tab](#)

Permite ver una página web dentro de Chrome a través del motor de Microsoft Internet Explorer.

#### [Web Developer](#)

La auténtica, la barra de herramientas para Mozilla Firefox, también está disponible para Google Chrome. Añade un menú y una barra de herramientas con numerosas opciones que ayudan a desarrollar páginas web. Además, tal como se explica en [Evaluación de la accesibilidad de sitios web con la barra de](#)



[herramientas de Firefox Web Developer Toolbar](#), este complemento se puede emplear para revisar la accesibilidad de un sitio web.

## **Microsoft Internet Explorer**

### [CheckIE](#)

La versión para Internet Explorer de [Checky](#). Reune diversos servicios de evaluación y reparación disponibles en línea que facilitan a los diseñadores y desarrolladores de sitios web la revisión de la calidad de sus trabajos. Por ejemplo, se puede validar el etiquetado de una página o su hoja de estilos CSS con las especificaciones del W3C, revisar su nivel de accesibilidad mediante TAW o Bobby, y todo desde el propio menú del navegador, sin tener que dirigirse al sitio específico de cada servicio.

### [DebugBar](#)

Contiene las siguientes utilidades: DOM Inspector (visualiza y permite modificar etiquetas y atributos de CSS), HTTP Inspector (visualiza las peticiones HTTP/S, las cookies y los parámetros pasados por GET y POST), Javascript Inspector y Javascript Console, HTML Validator y otras funciones. Es gratuito para uso personal.

### [Internet Explorer Developer Toolbar](#)

Proporciona un conjunto de herramientas para validar la sintaxis del HTML y el CSS de una página, explorar y modificar el DOM de una página, mostrar información sobre las imágenes de una página como sus dimensiones o su texto alternativo, visualizar y borrar las cookies de una página, etc.

### [Web Accessibility Toolbar 1.2](#)

Ayuda a inspeccionar las páginas web y comprobar si se cumplen los criterios de accesibilidad. Entre otras cosas, permite: identificar los componentes de una página web, facilita el uso de aplicaciones online, simula la "experiencia" de algunos tipos de usuarios (básicamente, usuarios con algún problema de visión) y proporciona enlaces a guías de referencia y otros recursos. La página [Toolbar and WCAG 1.0](#) explica cómo se puede emplear esta herramienta para comprobar los puntos de verificación de [WCAG 1.0](#).

### [Web Accessibility Toolbar 2.0](#)

Versión mejorada de [Web Accessibility Toolbar 1.2](#).

## **Mozilla Firefox**



La página web [Web Accessibility Testing Tools](#) ofrece una lista de complementos para Firefox clasificados como herramientas de evaluación de la accesibilidad web.

#### [Checky](#)

Valida y analiza documentos que contienen HTML, XHTML, CSS y otros lenguajes; comprueba el cumplimiento de las pautas de WCAG y Section 508.

#### [Colour Contrast Analyser](#)

Calcula el contraste de color y luminosidad de cada elemento de una página web. Más información: [Colour Contrast Analyser Firefox Extension](#).

#### [Complex Table Mark-up \(com tab\) Toolbar](#)

Ayuda a crear tablas accesibles: muestra los atributos headers e id en tablas complejas; crea el marcado para que una tabla sea accesible, tanto de forma manual como automática; crea una versión lineal de una tabla.

#### [Fangs - the screen reader emulator](#)

Crea una versión textual de una página web similar a cómo lee la página un lector de pantallas. Añade un menú contextual al navegador que abre una nueva ventana donde se puede ver la versión textual, la lista de encabezados y la lista de enlaces de la página.

#### [Firefox Accessibility Extension](#)

Permite analizar la accesibilidad de una página web y añade soporte de las nuevas características de WAI-ARIA.

#### [Fire Vox](#)

Extensión del navegador Mozilla Firefox que proporciona la capacidad de lector de pantalla.

#### [Glossary of Terms](#)

Crea un glosario de términos a partir de las abreviaturas y acrónimos que contiene una página web. Más información: [Glossary of Terms - Mozilla/Firefox Extension](#).

#### [HeadingsMap](#)

Crea un índice o mapa de los encabezados de una página web. Además, este complemento permite navegar por la página web saltando de un encabezado a otro, tal como lo hacen los usuarios que utilizan un [lector de pantallas](#).

#### [IE View](#)

Permite ver una página web dentro de Firefox a través del motor de Microsoft Internet Explorer.

#### [Juicy Studio Accessibility Toolbar](#)

Realiza tres funciones: permite examinar los roles y propiedades de las regiones activas de WAI-ARIA, permite examinar las tablas de datos y determina si el contraste de color es suficiente. Más información: [Examining WAI-ARIA Live Regions](#).

#### [Link Widgets](#)

Facilita la navegación entre páginas que forman una secuencia al añadir una barra de botones para ir a la primera, anterior, siguiente y última página. Para ello emplea la etiqueta <link> o lo adivina a partir del texto de los enlaces.

#### [Qompliance](#)

Permite acceder fácilmente a la herramienta de evaluación [Worldspace](#) que evalúa Section 508, WCAG 1.0 y WGAG 2.0. Más información: [Deque Systems Announces New Web Accessibility Plugin for Firebug](#).

#### [Table Inspector](#)

Visualiza la información sobre una tabla que suele estar oculta en los navegadores visuales: summary, headers, axis, scope y abbr. Más información: [Table Inspector - Mozilla/Firefox Extension](#).

#### [Textise](#)

Crea una versión textual de una página web.

#### [WAVE Toolbar](#)

Permite realizar las revisiones de [WAVE](#) sin tener que conectar con el servidor. Ofrece cuatro perspectivas de una página web: Errores, características y alertas; Estructura y orden; Sólo texto y Resumen.

#### [WCAG Contrast checker](#)

Comprueba si la combinación de dos colores es adecuada para su percepción visual en base a los requerimientos de [WCAG 1.0](#) y [WCAG 2.0](#).

#### [Web Developer](#)

Añade un menú y una barra de herramientas con numerosas opciones que ayudan a desarrollar páginas web. Además, tal como se explica en [Evaluación de la accesibilidad de sitios web con la barra de herramientas de Firefox Web Developer Toolbar](#), este complemento se puede emplear para revisar la accesibilidad de un sitio web.

#### [X-Ray](#)

Permite visualizar las etiquetas HTML de una página web sobre la misma página web, sin tener que ver el código fuente.

## 5. Plug-ins

Los plugins constituyen una forma de añadirle recursos y diversas funcionalidades al navegador. No son complementos, estos son aplicaciones o programas instalados, los plugins son solo pequeños archivos o librerías DLL que funcionan de enlaces entre el navegador y la aplicación original.

Por ejemplo si tienes instalado el plugin de Adobe PDF, al dar un clic en un documento PDF, en un sitio web determinado, se activará Adobe Acrobat que te permitirá ver el documento en el navegador. Generalmente se añaden a nuestro navegador al instalar las aplicaciones que los contienen, detectan su modelo y te agregan el plugin, no obstante tienes la posibilidad de deshabilitarlo si así te parece.

Algunos navegadores modernos al detectar que el contenido de la página necesita de ciertos plugins para mostrar algún contenido, ofrecen al usuario la posibilidad de instalarlos. Algunos son verdaderamente imprescindibles y necesarios para el aprovechamiento de la navegación web.

Puedes administrar los plugins que tienes instalados, eliminar los que no utilices, mantener actualizados los que te son necesarios e instalar los que no tienes y pueden serte útiles en la navegación. Accede a la configuración de los plugins de acuerdo al navegador que tengas de la siguiente forma:

- En Firefox y Chrome about:plugins o a través del menú Herramientas + Complementos. También tienes la opción de chequear tus plugins en la web de Mozilla.

<http://www.mozilla.com/en-US/plugincheck/>

- En Internet Explorer en el menú Herramientas + Administrar complementos + Barra de herramientas y extensiones, ahí los encontrarás mezclados con los complementos y otros add-ons.

### ***Los plugins más utilizados en los navegadores web***

#### **Shockwave Flash o Flash player**

Flash es un plugin para navegadores web que permite visualizar contenido y aplicaciones interactivas en internet, como juegos, animaciones, presentaciones y otros elementos multimedia e interactivos hechos para la web. Utiliza archivos con extensión SWF y FLV, cada nueva versión que sale de este producto hace que la carga de video sea más óptima y rápida, los efectos en el vídeo mejores y

más nítidos con disminución en el pixelado. Para aprovechar en la actualidad el amplio contenido multimedia que se ofrece en la web es imprescindible tener instalado el reproductor flash. El plugin para Adobe Flash Player viene incluido de forma predeterminada y con su versión más reciente en el navegador Google Chrome, en el resto de los exploradores de internet se hace necesaria su instalación manualmente antes de poder hacer uso de las aplicaciones flash. Muchas veces las aplicaciones flash embebidas en las páginas web, te permiten de forma automática instalar el plugin necesario para visualizarla, la otra opción es ir manualmente al sitio web del fabricante para instalar el plugin de flash.

## **Java**

Java es el plugin que habilita Java Runtime Environment (JRE), (no confundir con JavaScript) permite ejecutar en los navegadores los applets escritos en el lenguaje de programación Java, que son pequeñas aplicaciones incrustadas en el código HTML, las más comunes son juegos, animaciones, y diversas utilidades.

## **Microsoft Silverlight**

Silverlight es una estructura surgida recientemente del gigante Microsoft para aplicaciones web, que agrega nuevas funciones multimedia como la reproducción de vídeos, gráficos vectoriales, animaciones e interactividad, en forma similar a lo que hace Adobe Flash.

## **RealPlayer**

RealPlayer es un reproductor de multimedia, creado por RealNetworks, que reproduce varios formatos multimedia, incluyendo las generaciones múltiples de los códec de RealAudio y de RealVideo así como MP3, MPEG-4, QuickTime, etc.

## **QuickTime**

QuickTime permite el funcionamiento de una amplia variedad de contenido multimedia en las páginas web con una alta calidad, ya sea música, películas, videos, etc.