

# 1. Lenguajes de marcas

LM&SGI

Desarrollo de Aplicaciones Web

1. Redes de ordenadores .....	2
2. El problema del formato .....	4
3. HTML.....	5
4. Coexistencia de estándares .....	6
5. XML .....	7
6. DOCTYPE .....	7
7. HTML y CSS.....	8

## 1. Redes de ordenadores

### ⊙ Componentes principales.

Una red de ordenadores es un conjunto de equipos informáticos conectados entre sí, con el fin de compartir información y recursos.

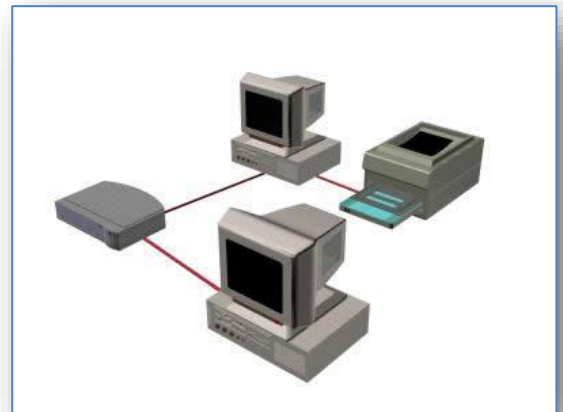
La red más pequeña que podemos formar necesita los siguientes componentes:

- **Dos ordenadores.** Son los equipos que van a compartir información o recursos( como un impresora). Según su actividad pueden cumplir dos roles:

**Ciente.** Es el ordenador que solicita información o el acceso a un recurso.

**Servidor.** Es el ordenador que provee información o el acceso a un recurso.

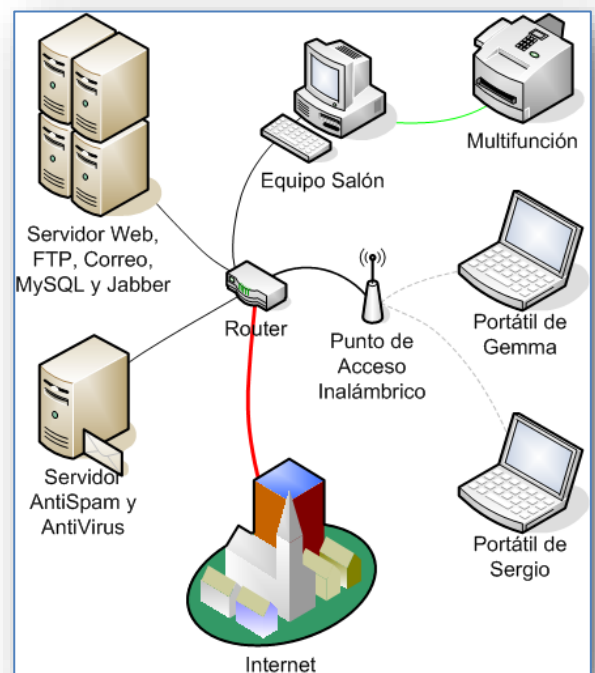
- Un **router**. Es el encargo de dirigir el tráfico de información entre ambos ordenadores.
- **Cable de red.** Sirve para unir los anteriores dispositivos.



Normalmente una red no se compone solo de dos ordenadores, sino que está formado por una gran cantidad. Eso no quiere decir que todos vayan a comunicarse con todos los otros, simplemente se brinda la posibilidad por si es necesario.

Una red de ordenadores también puede conectarse a otra red por medio de diferentes métodos, como por ejemplo unir sus routers. De esta forma una empresa de publicidad puede unir todos sus ordenadores uniendo las redes que tienen en diferentes edificios.

La idea de la interconexión de redes fue el germen que dio lugar a **Internet**, que es la interconexión a nivel mundial de todas las redes. Cuando accedemos a Internet desde nuestra casa, lo que estamos haciendo realmente es unirnos a la red del proveedor (como Telefónica). Haciendo esto nos estamos conectando a todos los ordenadores de Internet, lo que no quiere decir que todo el mundo nos de acceso a su ordenador.



### ☉ Funcionamiento de una petición de página web.

Supongamos que estamos en casa, tenemos un ordenador conectado a Internet y que queremos acceder a nuestro correo Web de Hotmail. Los pasos que se dan hasta que consigue acceder son los siguientes:

1. Abrimos el navegador de Internet y en la barra de direcciones escribimos la dirección: [www.hotmail.com](http://www.hotmail.com).
2. Los datos pasan del navegador al ordenador, después al router, navegan por Internet y llegan al equipo que hace de servidor de Hotmail.
3. Una vez aquí el servidor obtiene el documento que le están pidiendo y lo manda de vuelta al cliente.
4. El documento hace le trayecto inverso hasta llegar al equipo cliente.
5. El equipo cliente pasa el documento al navegador y este lo interpreta para mostrárselo al usuario final.



Cada vez que hacemos una petición en Internet se debe hacer este recorrido. En cada caso la respuesta dependerá del servidor al que estamos pidiendo datos.

### ☉ Protocolo HTTP

La transmisión de páginas web (que en definitiva son documentos HTML) se realiza mediante el protocolo http( HyperTextTransferProtocol), que es parte de la pila de protocolos TCP/IP. Se trata de un protocolo basado en una comunicación petición-respuesta; de modo que un cliente realiza una petición de recurso indicando su dirección, y un servidor responde a dicha petición bien transmitiendo al cliente el recurso solicitado o bien indicando un mensaje de error.

La dirección del recurso se indica utilizando la notación URL, que funciona así:

protocolo://servidor:puerto/rutaAlRecurso

En el caso de las páginas web, el protocolo es http. En la parte servidor, se indica la dirección del servidor (por ejemplo [www.jorgesanchez.net](http://www.jorgesanchez.net)) y la ruta es la ruta que hay que seguir por las carpetas y archivos del servidor para llegar al recurso.

Ejemplo:

<http://www.miweb.es/lenMarcas/primeraParte.html>

Permite al navegador mostrar en pantalla la página web `primeraParte.html` alojada en la carpeta `lenMarcas` del servidor [www.iesmaestredecalatrava.es](http://www.iesmaestredecalatrava.es) con el que se comunica utilizando el protocolo http (puesto que no se ha indicado puerto, se utilizará el puerto 80).

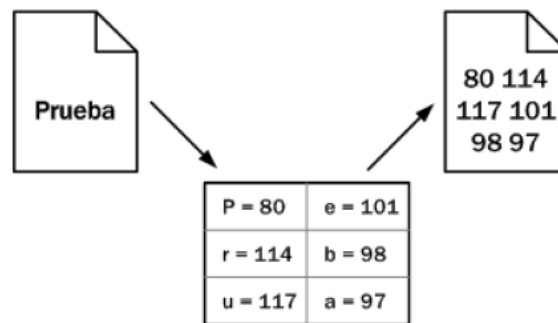
A veces no se indica la ruta y en ese caso el servidor envía la llamada página por defecto, página de inicio o home page que generalmente es un documento HTML llamado `index.html` o `default.html` o `home.html` y que se ubica en la carpeta raíz del servidor.

La comunicación entre el servidor y el cliente, normalmente se realiza a través del puerto 80. Pero se podría indicar otro puerto, ejemplo (usando el puerto 9123):

<http://www.miweb.es:9123/lenMarcas/primeraParte.html>

## 2. El problema del formato

Para almacenar un contenido de texto en un archivo electrónico, se utiliza una tabla de conversión que transforma cada carácter en un número. Una vez almacenada la secuencia de números, el contenido del archivo se puede recuperar realizando el proceso inverso.



El proceso de transformación de caracteres en secuencias de números se denomina codificación de caracteres y cada una de las tablas que se han definido para realizar la transformación se conocen con el nombre de páginas de código. Una de las codificaciones más conocidas (y una de las primeras que se publicaron) es la codificación ASCII.

Los caracteres propios de los idiomas diferentes al inglés también pueden ser problemáticos. El motivo es que desde que se crea una página web hasta que llega al navegador del usuario, intervienen numerosos procesos:

- ⊙ El diseñador crea la página web con su editor HTML (por ejemplo Dreamweaver).
- ⊙ Si se trata de una aplicación dinámica, el programador recorta la página HTML del diseñador y la mezcla con el resto del código de la aplicación (por ejemplo PHP).
- ⊙ El servidor web almacena las páginas HTML estáticas o el código de la aplicación web y sirve las páginas solicitadas por los usuarios.
- ⊙ El usuario solicita y visualiza las páginas web a través de su navegador.

Si el editor HTML del diseñador utiliza la codificación UTF-8, el entorno de desarrollo del programador también utiliza UTF-8, el servidor web sirve las páginas con esa codificación y el navegador del usuario es capaz de visualizar las páginas con formato UTF-8, el texto se verá correctamente en el navegador del usuario.

Sin embargo, muchas veces no es posible que todos los procesos involucrados utilicen la misma codificación de caracteres. Por limitaciones técnicas o por decisiones de los diseñadores y programadores, los textos pueden pasar de codificación UTF-8 a codificación ISO-8859 en cualquier momento. Si se produce este cambio sin realizar una conversión correcta, el navegador del usuario mostrará caracteres extraños en todos los acentos y en todas las letras como la ñ.

La solución más sencilla para asegurar que todos estos caracteres potencialmente problemáticos se van a visualizar correctamente en el navegador del usuario consiste en sustituir cada carácter problemático por expresiones llamadas entidades HTML. Por ejemplo:

&ntilde; → ñ  
 &Ntilde; → Ñ  
 &acute; → á  
 &eacute; → é

Una vez resuelto el problema de almacenar el texto simple, se presenta el reto del almacenamiento de los contenidos de texto con formato. En otras palabras, ¿cómo se almacena un texto en negrita? ¿y un texto de color rojo? ¿y otro texto azul, en negrita y subrayado? Utilizar una tabla de conversión similar a las que se utilizan para textos sin formato no es posible, ya que existen infinitos posibles estilos para aplicar al texto.

La solución que realmente se emplea para guardar la información con formato es mucho más sencilla: **el archivo electrónico almacena tanto los contenidos como la información sobre el formato de esos contenidos.**

El proceso de indicar las diferentes partes que componen la información se denomina **marcar (markup** en inglés). Cada una de las palabras que se emplean para marcar el inicio y el final de una sección se denominan **etiquetas**.

Aunque existen algunas excepciones, en general las etiquetas se indican por pares y se forman de la siguiente manera:

**<nombre\_etiqueta> ... </nombre\_etiqueta>**

La principal ventaja de los lenguajes de etiquetas es que son muy sencillos de leer y escribir por parte de las personas y de los sistemas electrónicos. La principal desventaja es que pueden aumentar mucho el tamaño del documento, por lo que en general se utilizan etiquetas con nombres muy cortos.

### 3. HTML

En los años 60 las empresas de publicación y manejo de documentos electrónicos tenían el problema de la falta de compatibilidad entre aplicaciones. El problema existente es que cada aplicación utilizaba sus propias marcas para describir los diferentes elementos.

IBM intentó resolver el problema con su lenguaje de marcas GML( Generalized Markup Language). Este lenguaje pasó a manos de ISO y se convirtió en SGML( Standard Generalized Markup Language)

SGML no es un lenguaje sino un metalenguaje, es decir, un conjunto de normas que permiten crear otros lenguajes de marcas. Esto se hace definiendo un vocabulario, una gramática y unos atributos concretos, pero siguiendo las normas de creación de SGML.

HTML es uno de los lenguajes creados a partir de SGML.HTML son las siglas de HyperText Markup Language (Lenguaje de marcación de hipertexto).



El origen de HTML se remonta a 1980, cuando el físico Tim Berners-Lee, trabajador del CERN (<http://www.cern.ch/>) (Organización Europea para la Investigación Nuclear) propuso un nuevo sistema de "hipertexto" para compartir documentos.

Los programas capaces de traducir el código HTML y producir una salida en pantalla de los mismos son los navegadores (browsers en inglés). Se trata de un software gráfico que se inició con la creación de Mosaic a principios de los 90 y que poco a poco produjo más productos hasta llegar a una guerra de navegadores a finales de los 90 entre Internet Explorer de Microsoft y Navigator de Netscape que ganó Microsoft, que ahora continúa con otros navegadores como el propio Internet Explorer, Mozilla Firefox (sucesor de código abierto de Netscape), Google Chrome, Opera o Apple Safari entre otros.

El problema surgió en cuanto unos navegadores incorporaron elementos HTML que el resto no traducía, con lo que aparecieron diferentes dialectos HTML. Así una página se podía mostrar de forma totalmente diferente según el navegador.

La solución pasó por intentar estandarizar el lenguaje. Por ello el propio Tim Bernes Lee fundó la World Wide Web Consortium (abreviado W3C) como organismo de estandarización del lenguaje HTML ante la industria. En la actualidad las directrices de W3C son seguidas por la mayoría de navegadores aunque no al 100%, lo que sigue generando problemas a los creadores de páginas web.

El lenguaje HTML es un estándar reconocido en todo el mundo. Como se trata de un estándar reconocido por todas las empresas relacionadas con el mundo de Internet, una misma página HTML se visualiza de forma muy similar en cualquier navegador de cualquier sistema operativo.

El lenguaje XHTML es una adaptación de HTML al lenguaje XML. XHTML añade una serie de restricciones al lenguaje para hacer que cumpla las normas impuestas por XML.

## 4. Coexistencia de estándares

Actualmente es posible utilizar cualquier de los siguientes lenguajes para crear una página web:

☉ **HTML 4.01.** Se trata de la versión estándar del HTML tradicional hecha en el año 1999 y que sigue teniendo mucha vigencia actualmente. Hay tres versiones: la transicional (que permite seguir usando algunas etiquetas que se consideran obsoletas), la estricta (que elimina numerosas etiquetas y atributos para forzar a crear un HTML con menos formato y más significado) y la frameset orientada a usar los ya muy poco utilizados marcos. La versión más popular es la transicional al ser más libre.

☉ **XHTML.** Era planteado como el sustituto de HTML, la primera versión, la 1.0 sigue siendo la más usada. Hay también versión estricta, transicional y frameset. La más utilizada, otra vez es la transicional porque permite el uso de numerosos atributos y elementos que se consideran obsoletos, pero que muchos diseñadores utilizan.

☉ **HTML 5.** Se considera el estándar desde 2012. Poco a poco se está convirtiendo en el HTML más utilizado. Se creó fuera del World Wide Web

Consortium en la entidad llamada WHATWG (Web Hypertext Application Technology Working Group), auspiciado por Apple y Mozilla entre otras empresas a las que se le han ido añadiendo muchas más. Aunque no es una recomendación oficial y por ahora los navegadores aceptan sólo partes de HTML 5.

## 5. XML

Es una simplificación y adaptación de SGML que permite definir lenguajes específicos. Por tanto, XML no es un lenguaje en particular, sino una manera de definir lenguajes, es decir, un metalenguaje.

XML no se usa solo en Internet sino que se está usando como un estándar en el intercambio de información estructura entre diferentes plataformas. Se suele usar en bases de datos ligeras, para evitar las restricciones de los SGBD relacionales.

## 6. DOCTYPE

El estándar XHTML deriva de XML, por lo que comparte con él muchas de sus normas y sintaxis. Uno de los conceptos fundamentales de XML es la utilización del DTD o *Document Type Definition* ("*Definición del Tipo de Documento*").

Un DTD es un documento que recoge el conjunto de normas y restricciones que deben cumplir los documentos de un determinado tipo. Si por ejemplo se define un DTD para los documentos relacionados con libros, se puede fijar como norma que cada libro tenga un título y sólo uno, que tenga uno o más autores, que la información sobre el número de páginas pueda ser opcional, etc.

El conjunto de normas, obligaciones y restricciones que se deben seguir al crear un documento de un determinado tipo, se recogen en su correspondiente DTD. El estándar XHTML define el DTD que deben seguir las páginas y documentos XHTML. En este documento se definen las etiquetas que se pueden utilizar, los atributos de cada etiqueta y el tipo de valores que puede tener cada atributo.

En realidad, la versión 1.0 del estándar de XHTML define tres DTD diferentes. Para indicar el DTD utilizado al crear una determinada página, se emplea una etiqueta especial llamada `doctype`. La etiqueta `doctype` es el único elemento que se incluye fuera de la etiqueta `<html>` de la página. De hecho, la declaración del `doctype` es lo primero que se debe incluir en una página web, antes incluso que la etiqueta `<html>`.

A continuación se muestran los tres DTD que se pueden utilizar al crear páginas web:

### ☉ HTML 4.01 Transicional

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
```

### ☉ XHTML 1.0 Transicional

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
```

Además la etiqueta raíz html debe usarse de este modo (especificando el espacio de nombres y el lenguaje de la página):

```
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="es">
```

### ☉ HTML 5

```
<!DOCTYPE html>
```

```
<html lang="es">
```

## 7. HTML y CSS

Originalmente, las páginas HTML sólo incluían información sobre sus contenidos de texto e imágenes. Con el desarrollo del estándar HTML, las páginas empezaron a incluir también información sobre el aspecto de sus contenidos: tipos de letra, colores y márgenes.

La posterior aparición de tecnologías como JavaScript, provocaron que las páginas HTML también incluyeran el código de las aplicaciones (llamadas scripts) que se utilizan para crear páginas web dinámicas. Incluir en una misma página HTML los contenidos, el diseño y la programación complica en exceso su mantenimiento.

Normalmente, los contenidos y el diseño de la página web son responsabilidad de diferentes personas, por lo que es conveniente separarlos. CSS es el mecanismo que permite separar los contenidos definidos mediante XHTML y el aspecto que deben presentar esos contenidos.

