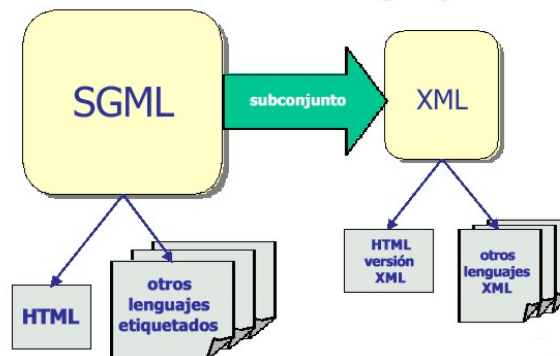


# Capítulo I

## Introducción



**Problemática actual en la Web**

**El proyecto GNU**

**W3C**

**Metadatos**

**Metalenguajes de etiquetas**

**Lenguajes de marcas**

**Comparación entre HTML, XML y SGML**

En la actualidad la Web tiene un gran número de usos: las empresas ponen a disposición del mundo información acerca de los servicios y/o productos que ofrecen, diferentes tipos de organismos informan acerca de sus actividades y establecen comunicación con las personas. Con la Web las organizaciones tienen la oportunidad de darse a conocer prácticamente a cualquier persona en el mundo a un costo bajo.

Una de las características de la Web que tiene más aceptación es el desarrollo de aplicaciones web. Las aplicaciones web, que se caracterizan por ser sitios web activos que permiten la modificación de sus contenidos y características por sus usuarios, tal como si fuera una aplicación de escritorio que se ejecutara en red, se las realiza a través de diferentes tecnologías tanto de hardware como de software, y tienen costos de licencias de uso muy variados: desde las propuestas desarrolladas con software gratuito hasta los modelos que usan software con costo de varios miles de dólares.

Con el tiempo el Software Libre ha tomado importancia en casi todos los ámbitos donde el software en general es utilizado. En la actualidad el software libre goza de mucha aceptación; por un lado, importantes empresas comerciales se han decidido a dar soporte a alguno de estos productos libres; por otro, los mejores proyectos de software libre cuentan con una amplia base de colaboradores, lo que lleva a que el software que finalmente ve la luz esté altamente probado e incluya las peticiones de una inmensa mayoría de usuarios.

### **1.1. Problemática actual en la Web**

El considerable crecimiento del WWW ha sido impulsado por la posibilidad de distribuir de forma fácil y barata documentos electrónicos a una audiencia internacional. Como estos documentos se hacen cada vez más grandes y complejos, los usuarios han comenzado a experimentar las limitaciones de un medio que no proporciona la extensibilidad, estructura y comprobación de datos necesarios para la publicación a gran escala.

Cada día se demandan más servicios a través de Internet y aumenta el número de competidores, todo este dinamismo requiere sistemas potentes en cuanto a servicios y contenidos, otro problema es el presentar información en diversos dispositivos y formatos, una empresa puede presentar los datos de un producto visualizados desde un navegador web, desde un móvil con tecnología WAP o que se pudiesen imprimir con alta calidad para un catálogo en papel.

El Internet a más de tener un estándar de acceso y presentación de información, debe establecer un estándar para la comprensión de la información, una forma común de representar los datos para que el software pueda buscar, desplazar, presentar y manipular mejor los datos.

HTML ofrece amplias facilidades de presentación, es fácil de aprender, sin embargo es demasiado superficial, además es estático, no se puede modificar ni adaptar a las necesidades particulares de los desarrolladores.

HTML, a pesar de los parches que se le han ido poniendo, se ha quedado corto para las nuevas necesidades de la Web, HTML sólo permite el formateo de documentos de acuerdo a unas reglas preestablecidas, con él es imposible definir marcadores propios para aplicaciones específicas.

Para contrarrestar la problemática de la Web se requiere de un estándar de representación de datos que ampliaría Internet, este estándar sería el vehículo para las transacciones comerciales, la publicación de perfiles de preferencias personales, la colaboración automatizada y el uso compartido de bases de datos. El estándar de datos permitirá una gran variedad de usos nuevos, todos basados en una representación estándar para el desplazamiento de datos estructurados por toda la Web. Este estándar de datos es XML (Lenguaje extensible de marcas) y sus extensiones, cuyo uso permitirá que los sitios en Internet mejoren sus características y rendimiento, gestionando su información de forma ágil, oportuna y estructurada; desarrollando aplicaciones donde se puede controlar mucho mejor el ambiente.

## **1.2. El proyecto GNU**

### **1.2.1. La Free Software Foundation y el proyecto GNU**

La Free Software Foundation (Fundación para el software libre) promueve el desarrollo y la utilización del Software Libre a todos los niveles, de manera específica ha lanzado un proyecto que consiste en el desarrollo de un sistema completo de Software Libre llamado GNU. Este sistema comprende desde el Sistema Operativo hasta la aplicación más insignificante

### 1.2.2. Software Libre

El término Software Libre se refiere a la libertad de los usuarios de ejecutar, copiar, redistribuir, estudiar, cambiar y mejorar el software. Como «free» [libre] se refiere a libertad y no a precio, no existe contradicción entre la venta de copias y el software libre.

En el caso del Software Libre lo importante es:

1. La redistribución libre.
2. La disponibilidad de las fuentes.
3. La posibilidad de modificación.
4. La no discriminación entre comercial y gratuito.

### 1.2.3. Ventajas del Software Libre

- **Es Adaptable.**- La disponibilidad del código fuente permite añadir aquello que interese o eliminar lo que sobre. De la misma manera, se pueden usar librerías desarrolladas por otros en programas propios.
- **Es portable.**- La mayoría de fabricantes de Software sólo desarrollan para las plataformas más utilizadas del mercado (Microsoft Windows), esto obliga a utilizar estas plataformas para disponer de ciertos paquetes informáticos. Disponer del código fuente permite la recompilación del software sobre nuevas plataformas, permitiendo que se use el sistema operativo que mejor cubra las necesidades.
- **Evoluciona más rápido.**- Los conceptores de Software Libre reciben la ayuda de los usuarios, quienes les transmiten las sugerencias sobre posibles mejoras. La mayoría de proyectos de Software Libre son desarrollados por equipos de personas distribuidos por el mundo y que se comunican por Internet.

### 1.2.4. Inconvenientes del modelo de software libre

Los críticos del software libre afirman que este modelo necesita un grupo enorme de programadores que sean benevolentes y que tengan un deseo real de crear un producto fiable y seguro y si este grupo no es benevolente entonces el modelo falla. Sin embargo la historia muestra que las personas que se vinculan con este movimiento y en particular con Linux son benevolentes, su objetivo final es obtener software de alta calidad y un reconocimiento.

### 1.3. Consorcio W3C - World Wide Web Consortium

El W3C fue fundado en octubre de 1994 con el objetivo de desarrollar protocolos comunes para la evolución de Internet. Se trata de un consorcio de la industria internacional con sedes conjuntas en el Instituto Tecnológico de Massachussets (EEUU), en el Instituto Nacional de Investigación en Informática y Automática (Europa) y la Keio University Shonan Fujisawa Campus (Japón).

El W3C es un consorcio industrial que busca promover estándares para la evolución del Web y la interoperabilidad entre productos para el WWW al producir especificaciones y software de referencia. Tiene como misión la publicación de protocolos o estándares globales de uso libre. Dichas especificaciones están disponibles sólo para los miembros del forum durante su estado de desarrollo, hasta que son publicadas y quedan a disposición de todo el mundo para su uso público, ya sea comercial o no.

En el W3C, se puede encontrar:

- Un repositorio de información acerca del WWW para desarrolladores y usuarios.
- Código de referencia para la implementación de los distintos estándares.
- Prototipos diversos y aplicaciones de ejemplo para demostrar el uso de nuevas tecnologías fomentadas por el W3C.

El W3C consta de una colección de grupos interesados que se reúnen para discutir acerca de nuevas tecnologías en Internet. Su proceso es bastante considerable y profundo, y tiene como resultado una decisión consensuada, de forma que el resultado final tenga gran impulso.

#### 1.3.1. Proceso de trabajo del W3C

Primero alguien (un grupo, compañía, persona) eleva una nota acerca de una tecnología o un asunto técnico al W3C. Después de un debate y consulta, se crea una propuesta de borrador. Con suerte y mucho trabajo, se puede llegar a una recomendación. En este momento, la tecnología se considera lista para su adopción en la industria.

Etapas en el desarrollo de una especificación:

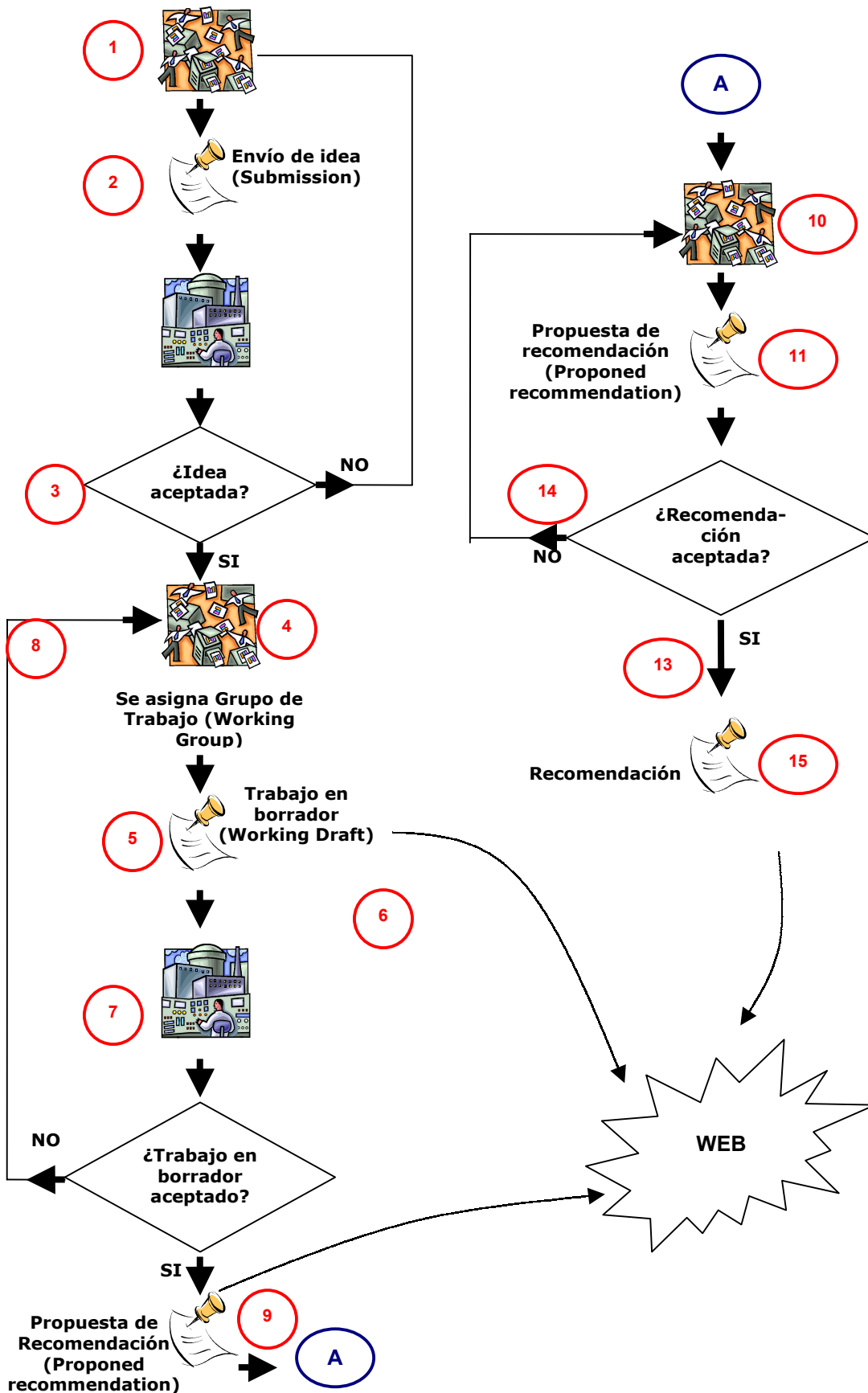
- **Nota (Notes).**- Una nota es simplemente una idea, no representa una obligación del W3C para desarrollar un trabajo relacionado con el contenido en ésta.
- **Envío (submission).**- Un envío es una propuesta de trabajo planteada por uno o varios miembros del Consorcio para que se someta a consideración, y para que

se le asigne un grupo de trabajo que comience a trabajar sobre la idea. Cuando a una idea se le asigna un grupo de trabajo, es cuando los siguientes documentos que se publiquen serán trabajos preliminares, y en ocasiones se publican algunas notas relacionadas al tema.

- **Borrador de trabajo (Working drafts).**- Este es un trabajo en progreso, un grupo dentro del W3C está activamente trabajando en él. Estos documentos se publican cada vez que son rechazados después de intentar promoverlos como una Propuesta de Recomendación, por lo que a pesar de que se publican en el Web, pueden tener aun cambios considerables.
- **Propuesta de Recomendación (Proposed Recommendation).**- Algún consenso ha sido logrado dentro del grupo de trabajo. Se ha elevado para revisión. Una especificación en esta etapa, aún puede tener cambios, pero mínimos comparados con los que sufre en el Trabajo Preliminar.
- **Recomendación (Recommendation).**- La idea que comenzó como una nota ahora se ha vuelto 'oficial'. Según el W3C, esta tecnología está lista para su uso.

El W3C no impone los acuerdos hechos o las tecnologías diseñadas. Cada una de las partes involucradas puede "extender" o mejorar los diseños.

A continuación se muestra un diagrama en el que se representa el paso de una especificación desde el momento que surge como idea y se envía al W3C, hasta el momento en que es aceptada y publicada como una Recomendación.



1. Una idea se manifiesta por uno o varios miembros del Consorcio, y se trabaja en ésta para proponerla como un nuevo estándar.
2. La idea se envía a consideración de los miembros del consorcio.
3. Si la idea se considera relevante, entonces se crea un Grupo de Trabajo (Working Group) que a partir de ese momento comenzará a pulir la especificación de la nueva tecnología. Una idea puede ser rechazada por falta de relevancia o de originalidad, la cual puede ser retomada, o descartada.
4. y 5. El Grupo de Trabajo formado principalmente por los miembros que propusieron la idea, comienza a publicar en el Web (6) los Trabajos Preliminares (Working Drafts) de la nueva tecnología, de manera que comienzan a tener retroalimentación de personas ajenas al Consorcio.
7. Cada determinado tiempo, los avances se vuelven a pasar a consideración del Consorcio, para decidir su posible aprobación como “Propuesta de Recomendación”
8. Si la especificación no cumple con lo esperado, es rechazada, para corregir y seguir trabajando en la versión preliminar y se repite el ciclo hasta que...
9. El trabajo preliminar se vuelve Recomendación Propuesta, la cual a su vez regresa al Grupo de Trabajo (previa publicación en el Web) para que continúen con las correcciones y sigan teniendo retroalimentación de las personas ajenas al Consorcio por medio de listas de discusión y de listas de correo que el mismo Consorcio administra.
10. y 11. El Grupo de Trabajo continúa realizando modificaciones y actualizaciones a las especificaciones, para mandarlas nuevamente a consideración del ...
12. Consorcio, para que sean evaluadas para su posible aprobación como Recomendaciones.
13. Si la Recomendación propuesta cumple con lo esperado, el trabajo se vuelve Recomendación (15), y se publica como especificación final en el Web.
14. Si la Recomendación Propuesta se rechaza, el Grupo de Trabajo sigue modificándola, y sigue el ciclo hasta que logra volverse Recomendación (15).



## 1.4. Metadatos

La enorme cantidad de recursos de información disponibles en Internet, la disparidad en las calidades de los contenidos y la dificultad de encontrar información relevante con cierta rapidez y eficiencia, pusieron de manifiesto la necesidad de establecer un mecanismo para describir los datos contenidos en la Web, con el objetivo principal de hacer más efectiva la recuperación de la información en Internet, la respuesta a esta necesidad son los metadatos.

Los Metadatos son definidos como "Datos acerca de los datos" o "información acerca de la información". Los metadatos consisten en información que caracteriza los datos. Los metadatos son utilizados para suministrar información sobre datos producidos, el metadato describe el contenido, la calidad, condición y otras características de un dato. Por ejemplo, el catálogo de una biblioteca es un registro de metadatos, en el sentido de que describen publicaciones.

El W3C ha comenzado a centrarse en la configuración de una arquitectura de metadata para el Web. El Resource Description Framework o RDF (Infraestructura para la Descripción de Recursos), que se está desarrollando para sustentar diversas necesidades de metadatos que tienen proveedores y vendedores de información.

Entre las aplicaciones que pueden beneficiarse de los metadatos, también conocidos como metacontenido, se encuentran:

- Indización de sitios Internet o Intranet.
- Sistemas de evaluación de contenido, definición de canales de distribución automática de contenidos, firmas digitales.
- Recogida de datos de los motores de búsqueda, catalogación de bibliotecas digitales y personalización de contenido de otros fabricantes.
- RDF también se puede utilizar para describir diferentes tipos de información, como la que se encuentra en sitios Web, en archivos, en el escritorio de un usuario, segmentos de correo electrónico o bases de datos ya existentes.
- Mapas de los sitios para facilitar la navegación.
- Integrar páginas Web y archivos locales de forma transparente, y facilitar la navegación y búsqueda de información.

### **1.4.1. RDF(Resource Description Framework) - Infraestructura para la Descripción de Recursos**

RDF es parte de la Actividad del W3C en materia de metadatos. La finalidad de esta actividad, y específicamente de RDF, es producir un lenguaje para el intercambio legible por máquina, de descripciones de recursos en la Web. <sup>[WWW001]</sup>

RDF es una base para procesar metadatos; proporciona interoperabilidad entre aplicaciones que intercambian información en la Web. RDF resalta las facilidades para hacer posible el procesamiento automático de los recursos web. <sup>[WWW001]</sup>

RDF puede utilizarse en distintas áreas de aplicación; por ejemplo: en recuperación de recursos para proporcionar mejores prestaciones a los motores de búsqueda, en catalogación para describir el contenido y las relaciones de contenido disponibles en un sitio Web, una página Web, o una biblioteca digital particular, por los agentes de software inteligentes para facilitar el intercambio y para compartir conocimiento; en la calificación de contenido, en la descripción de colecciones de páginas que representan un documento lógico individual, para describir los derechos de propiedad intelectual de las páginas web, y para expresar las preferencias de privacidad de un usuario, así como las políticas de privacidad de un sitio web.

## **1.5. Metalenguajes de etiquetas**

Son lenguajes que sirven para definir otros lenguajes, cuyo objetivo es llevar a cabo la estructuración de textos mediante un conjunto de etiquetas, de tal manera, que puedan ser entendidos por los humanos y también procesados por los computadores. Estos lenguajes tienen auge sobre Internet, se los usa para la creación de documentos, y el intercambio o transferencia de información.

### **1.5.1. GML (General Modeling Language) - Lenguaje de modelado general**

En Internet donde convergen cantidad de idiomas, tecnologías, lenguajes; nació la necesidad de buscar un factor común, "un estándar", para resolver problemas asociados al tratamiento de documentos en diferentes plataformas.

El principal problema era que cada aplicación utilizaba sus propias marcas para describir los diferentes elementos. Las marcas son códigos que indican a un programa cómo debe tratar su contenido y así, si se desea que un texto aparezca con un formato determinado,

dicho texto debe ir delimitado por la correspondiente marca que indique como debe ser mostrado en pantalla o impreso.

Conociendo a la perfección el sistema de marcas de cada aplicación sería posible pasar información de un sistema a otro sin necesidad de perder el formato indicado. La forma que IBM creó para solventar esto se basaba en tratar las marcas como texto accesible desde cualquier sistema (texto plano, código ASCII), esta norma se denominó GML (General Modeling Language). Más tarde, GML cayó en manos de ISO, que lo convirtió en un estándar oficial denominándose SGML, esta norma de carácter general se aplica desde entonces para diseñar lenguajes de marcas específicos.

### **1.5.2. SGML (Standard Generalized Markup Language) - Lenguaje de marcas generalizado estándar**

SGML es el progenitor de todos los lenguajes de marcado, fue diseñado para permitir el intercambio de información entre distintas plataformas, soportes físicos, lógicos y diferentes sistemas de almacenamiento y presentación (bases de datos, edición electrónica, etc.).

Debido a que empresas y organizaciones necesitan compartir datos con un gran número de socios, clientes y diferentes departamentos internos, necesitan documentos que sean independientes de la aplicación. SGML se diseñó para satisfacer esta necesidad, es un lenguaje de marcado completamente independiente de cualquier aplicación.

SGML es una norma ISO derivada de una anterior (GML de IBM). SGML permite que la estructura de un documento pueda ser definida en base a la relación lógica de sus partes. Esta estructura puede ser validada por una Definición del Tipo de Documento (DTD - Document Type Definition). La norma SGML define la sintaxis del documento y la sintaxis y semántica de DTD.

Un documento SGML se marca de modo que no dice nada respecto a su representación en la pantalla o en papel. Un programa de presentación debe unir el documento con la información de estilo a fin de producir una copia impresa en la pantalla o en el papel.

SGML utiliza una Definición del Tipo de Documento (DTD – Document Type Definition) para determinar la estructura del documento. La DTD especifica los elementos y atributos que pueden utilizarse en el documento y qué caracteres se van a utilizar para marcar el texto. En SGML se pueden utilizar paréntesis (<>), guiones (-) o cualquier otro carácter para marcar un documento, siempre que dicho carácter especial se defina en la DTD.

A un conjunto de elementos de propósito general, utilizado para describir un determinado tipo de documento, se le denomina aplicación SGML. Una aplicación SGML también incluye reglas que especifican la manera con que los elementos podrán ser organizados.

SGML puede parecer el lenguaje extensible perfecto para describir documentos web. Sin embargo, los miembros del W3C que analizaban estos temas consideraban que SGML era demasiado complicado e incómodo para suministrar de manera eficiente información en la Web. La flexibilidad y la fluidez de las funcionalidades proporcionadas por SGML harían difícil crear el software necesario para procesar y visualizar la información SGML en los navegadores web. Lo que hacía falta era un subconjunto moderno de SGML, diseñado específicamente para proporcionar información en la Web. En 1996, el Grupo de Trabajo XML de W3C desarrolló este subconjunto, al cual denominaron XML (Lenguaje extensible de marcas).

### 1.5.3. XML - Lenguaje extensible de marcas

A partir del metalenguaje SGML, el W3C desarrolló otro lenguaje con algunas de las características más importantes de SGML, éste es el **XML** (**eXtensible Markup Language**) que es un metalenguaje para crear otros lenguajes de etiqueta. Es decir es una norma que define la sintaxis y las reglas a seguir para crear un lenguaje de etiquetas específico.

XML se especializa en la **gestión de información** para la Web, ya que posee extensibilidad y flexibilidad permitiendo agrupar una variedad amplia de aplicaciones, desde páginas web hasta bases de datos.

XML se usará con dos fines principales: el intercambio de datos entre las computadoras y seres humanos y el intercambio de datos entre las propias computadoras. En ambos casos se lleva a cabo el mismo procedimiento: una base de datos contiene la información de origen, una máquina intermedia recibe y procesa esos datos, y por último la máquina cliente muestra o introduce la información. Con XML además será posible crear páginas con distintos formatos en la máquina de destino: para mostrar en pantalla, sólo texto, para impresión, y todo ello con el mismo documento XML de origen.

A continuación explicamos de manera breve ciertos términos relacionados con XML:

**DTD** (Document Type Definition): Definición del tipo de documento. Una DTD es similar al esquema de una base de datos: define y asigna nombres a los elementos que se podrán utilizar en el documento, el orden en que dichos elementos podrán aparecer, los atributos de los elementos. Una DTD es un archivo/s que encierra una definición formal de un tipo

de documento y especifica la estructura lógica de cada documento. En XML no existen DTDs predefinidas, es el propio diseñador quien las crea según sus necesidades.

**XSL** (eXtensible Stylesheet Language): Lenguaje extensible de hojas de estilo. Define o implementa el lenguaje de estilo de los documentos escritos para XML. Permite especificar cómo se desea presentar los datos de un documento XML, también sirve para filtrar los datos de acuerdo a ciertas condiciones. Las hojas de estilo sirven para dar formato a la información, es la forma de poder visualizar el contenido de los documentos XML en distintos formatos. A un mismo documento XML se le pueden aplicar las hojas de estilo que se quiera e incluso se puede transformar un documento XML en otro distinto mediante XSLT (Transformaciones XSL).

**XML Linking Language** (XLink, Lenguaje de vinculación XML): Permite vincular documentos XML. Posibilita la existencia de múltiples destinos vinculados y otras funcionalidades avanzadas y se considera más potente que el mecanismo de vinculación de HTML.

**XML Pointer Language** (XPointer, Lenguaje de puntero XML): Permite definir destinos vinculados flexibles, se puede utilizar XPointer junto a XLink y crear vínculos a cualquier lugar del documento de destino, no solamente a un determinado destino previamente definido.

**DOM** (Document Object Model): Modelo de Objetos de Documento. Es una especificación que define como procesar documentos XML. El DOM para XML es un modelo de objetos que muestra el contenido de un documento XML.

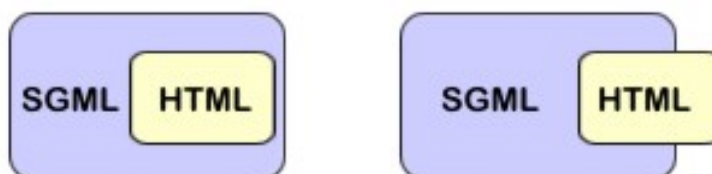
## 1.6. Lenguajes de marcas

Son sistemas complejos de descripción de información, normalmente documentos, que si se ajustan a la norma SGML. Las marcas más utilizadas suelen describirse por textos descriptivos encerrados entre signos de "menor" (<) y "mayor" (>), siendo usual que existan una marca de principio y otra de final.

### 1.6.1. HTML (Hypertext Markup Language) - Lenguaje de marcas de hipertexto

HTML, es un lenguaje simple que se utiliza para crear documentos de hipertexto para WWW. El formato de los documentos se marca mediante etiquetas (tags) que indican el comienzo y el final de los elementos que componen el documento. Cada uno de estos elementos tiene un significado estructural diferente.

El lenguaje HTML es originariamente un subconjunto del más completo SGML, especializado en la descripción de documentos en pantalla a través de marcas (*tags*, etiquetas). El proyecto inicial se basaba en una colección de etiquetas que permitían describir documentos de texto y vínculos de hipertexto que habilitaban desplazarse entre diferentes documentos, siempre con independencia de la máquina.



**Figura 1.1.** Situación inicial y actual de HTML con respecto a SGML.

La facilidad de uso y la particularidad de que no es propiedad de nadie, hizo al HTML el sistema idóneo para compartir información en Internet. La expansión de Internet le ha dado una posición de privilegio y ha hecho que la idea inicial se modifique considerablemente.

En principio, la intención de HTML era que las etiquetas fueran capaces de marcar la información de acuerdo con su significado, sin importar cómo se mostraban en la pantalla. Lo importante era el contenido y no la forma, o sea, que era un lenguaje de marcas orientado a describir los contenidos.

Por diversos motivos, los creadores de los navegadores fueron añadiendo más etiquetas HTML dirigidas a controlar la presentación y los usuarios las utilizaron para que sus documentos estuviesen perfectamente formateados, por lo que HTML pasó a ser un lenguaje de marcas más dirigido al control de la presentación.

Si a esto le añadimos que para facilitar la vida a los usuarios, los analizadores sintácticos de las marcas que incluyen los navegadores permitieron saltarse algunas normas sin que el propio usuario lo notase, dando como resultado que HTML ya no es un lenguaje que sigue las normas estrictas del SGML.

Llegados a un punto en el que HTML dejó de servir para su función inicial, el Consorcio World Wide Web (W3C) ha creado un nuevo subconjunto del SGML que sirve para describir contenidos de documentos, al que ha denominado XML.

### 1.6.2. XHTML

XHTML es un lenguaje que permite definir HTML como aplicación de XML. El XHTML 1.0 es exactamente igual que HTML 4.0 excepto en que sigue las reglas de XML. Para ello, utiliza las reglas de las DTDs (Declaraciones de Tipo de documento) definidas por el W3C.

Por ejemplo, en HTML el elemento <p> no necesita la etiqueta final </p>, en XHTML sí: todos los elementos necesitan una etiqueta inicial y otra final. En XHTML, las etiquetas tienen que ir obligatoriamente en minúsculas.

Las ventajas más evidentes que ofrece el migrar a XHTML son:

- Los documentos XHTML se establecen en base a las reglas XML. Por tanto, pueden ser visualizados, editados y validados por cualquier herramienta XML.
- Los documentos XHTML pueden funcionar tan bien (ó mejor) que los documentos HTML 4.0 en los navegadores existentes, así como en los navegadores que trabajen con XHTML.
- Los documentos XHTML pueden contener aplicaciones (applets o scripts) que se basen en DOM (Modelo de objetos de documento) y que modifiquen la estructura del documento XHTML.
- Permiten insertar en el documento XHTML marcas propias que no tienen por qué estar definidas en el estándar general. Esto es lo que se llama modularización XHTML.
- Hacer las páginas legibles por personas discapacitadas, es mucho más fácil, construir agentes de usuario que lean el contenido a personas invidentes o lo pasen a otros formatos.

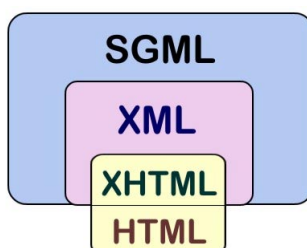
### 1.7. Comparación entre HTML, XML Y SGML

En esta sección se comparan los lenguajes XML, HTML, y SGML y sus características.

XML no es un nuevo lenguaje que vaya a suplantar a HTML. En realidad es una versión de SGML, más sencilla y más fácil de aplicar que SGML, diseñada para hacer frente a los problemas de compatibilidad y adaptabilidad de las nuevas tecnologías de Internet.

XML es un subconjunto de SGML, todos los documentos XML se ajustan a las normas de SGML. En teoría, HTML es un subconjunto de XML, especializado en presentación de

documentos para la Web, mientras que XML es un subconjunto de SGML especializado en gestión de información para la Web. En la práctica, HTML está un poco dentro de XML (que a su vez es parte de SGML) y otro poco fuera de SGML. Para reconducir esto, el grupo W3C ha dictado reglas expresas para distinguir el HTML que sigue al pie de la letra las normas de XML, denominándolo XHTML (eXtensible HyperText Markup Language), que no es más que una reformulación de HTML 4 dentro de las normas de XML.



**Figura 1.2.** Relación entre lenguajes de marcas.

A continuación se presenta un cuadro comparativo de HTML, XML, y SGML:

	HTML	XML	SGML
<b>Gramática</b>	Fija y no ampliable	Extensible	Extensible
<b>Estructura</b>	Monolítica	Jerárquica	Jerárquica
<b>Nº de marcas</b>	Fijas	Sin límite	Sin límite
<b>Complejidad</b>	Baja	Mediana	Alta
<b>Diseño de páginas</b>	Fijado por tags. Etiquetas con atributos CSS en DHTML	CSS o XSL	DSSSL
<b>Enlaces</b>	Simples enlaces	Poderosos enlaces	HyTime
<b>Exportabilidad (formatos/aplicaciones)</b>	No	Sí	Sí
<b>Validación</b>	Sin validación	Pueden validarse	Obligatorio DTD



	HTML	XML	SGML
<b>Búsquedas</b>	Simple y a veces resuelta por scripts o CGI	Es posible potente búsqueda. Con capacidad para personalizarla	Son posibles potentes búsquedas.

**Tabla 1.1.** Características de HTML, XML y SGML

**Gramática.** HTML es una aplicación de SGML, cuyo objetivo es la presentación de la información en pantalla, en HTML la gramática es fija, es imposible la creación de etiquetas personalizadas acorde a las necesidades que se presenten. XML es un metalenguaje, al igual que SGML, que tiene la característica de ser extensible, es decir con él se puede crear un propio lenguaje de marcado.

**Estructura.** En HTML la estructura es monolítica, mientras que en XML y SGML la estructura es Jerárquica. Todo el trabajo que se ha adelantado en SGML se puede convertir fácilmente al formato XML sin pérdida de información o características, lo cual no ocurre en el caso de querer convertirlo a HTML, donde sí se perderían bastante características útiles.

**Número de marcas.** En HTML se utilizan etiquetas predefinidas especificadas en DTDs; mientras que XML al igual que SGML permite que esas etiquetas sean definidas por el creador de la página, dando la posibilidad de que se puedan definir un conjunto ilimitado de etiquetas. La principal ventaja de XML sobre SGML es que soporta tags y estilos, lo cual le permite poder viajar por la Web.

**Complejidad.** HTML ha llegado a ser tan popular debido a que es un lenguaje fácil de aprender; XML es un lenguaje más prolijo, una marca olvidada o un valor de atributo sin comillas convierten el documento en inutilizable. SGML es un metalenguaje más complejo que los otros dos, es por ello que nació la idea de simplificarlo y adecuarlo para su uso en Internet a través de la creación de XML.

**Diseño de páginas.** Para diseñar una página en HTML se utilizan etiquetas, cuyo objetivo es definir el aspecto para su posterior presentación en pantalla, al HTML se han ido adhiriendo nuevas características que buscan hacer de él un lenguaje más potente. Una página diseñada en XML consta de un documento XML, que contiene información que para ser presentada requiere de una hoja de estilo (a través de la cual se da formato a XML) que puede ser una CSS (Hoja de estilo en cascada) o una hoja de estilo XSL

(creada mediante XSL- Lenguaje extensible de hojas de estilo). SGML utiliza DSSSL (es un estándar mediante el cual se transforma y formatea documentos SGML).

**Enlaces.** eXtensible Link Language (XLL), es el estándar para hacer enlaces empleado por XML. Permite enlaces bidireccionales y enlaces indirectos. Los enlaces en HTML son simples. SGML utiliza Hytime (es una norma internacional para la creación de enlaces de hipertexto y sincronización de información estática y dinámica en documentos múltiples convencionales y multimedia).

**Exportabilidad.** HTML no permite exportar los documentos a otro formato de presentación debido a que sus etiquetas definen el estilo de presentación para un navegador web. XML al igual que SGML permite la exportabilidad a otros formatos de publicación por ejemplo: papel, web, cd-rom, etc.; en el caso de XML esto se logra a través de XSL y XSLT. XML permite acceso a los datos para intercambio y no sólo para visualización como ocurre con HTML.

**Validación.** HTML posee una gramática fija, en la cual no requiere de validación, pues se utilizan etiquetas predefinidas. En XML es posible definir DTD's que indican qué etiquetas se pueden usar en un documento, en qué orden deben aparecer, cuáles pueden aparecer dentro de otras, cuáles tienen atributos, etc. Es decir podemos validar nuestro documento. En SGML el uso de las DTD's es obligatorio.

**Búsquedas.** Las búsquedas en HTML se resuelven a través de scripts, éstas son simples y los resultados devueltos no son siempre los deseados. Con XML se posibilitará realizar búsquedas que devuelvan respuestas más adecuadas y precisas. SGML, también posee la característica de permitir desarrollar búsquedas potentes.

### 1.7.1. XML y HTML

XML no es una nueva versión de HTML, aunque los dos proceden de un mismo metalenguaje, el SGML. XML es un metalenguaje, dado que con él se pueden definir lenguajes específicos de marcas y a diferencia del HTML, que se centra en la representación en la pantalla de la información, XML se centra en la información en si misma. El objetivo del desarrollo del XML, es que éste sea un estándar que sustituya a todo el conjunto de tecnologías que actualmente permiten acceder a la información a través del web. XML también utiliza etiquetas como SGML/HTML, sin embargo, a diferencia de HTML que ya posee DTD's específicos, en XML es posible describir información general como productos, descripciones, nombres...etc, los cuales son denominados vocabularios.

XML es diferente de HTML en tres puntos mayores:

1. Los proveedores de información pueden definir nuevos nombres de elementos y de atributos a su voluntad.
2. Las estructuras de los documentos pueden ser entrelazadas en cualquier nivel de complejidad.
3. Todo documento XML puede contener una descripción facultativa de su gramática para utilización entre aplicaciones que tengan necesidades de hacer la validación estructural.

HTML sigue siendo el principal lenguaje utilizado para indicar a los navegadores cómo representar la información en la Web, sin embargo presenta ciertas carencias, pues está limitado en:

- Inteligencia: Los conceptos no pueden conocerse a si mismos. Pese a tener cierta inteligencia, sabe lo que es un párrafo o una imagen, pero no puede saber sobre que trata el párrafo o que imagen es. El poder de la inteligencia no existe.
- Adaptabilidad: No facilita la adaptación.
- Mantenimiento: Su mantenimiento se hace complicado.

El XML tiene todas las características que hicieron grande al HTML:

- Asociabilidad: Al igual que HTML, puede realizar asociaciones simples de conceptos, pero además ofrece asociaciones complejas.
- Simplicidad: Aunque deba sacrificar algo de simplicidad a favor de potencia, este cambio es rentable para el XML.
- Portabilidad: Puede ser portado cómodamente.

Pero además ofrece características que el HTML nunca podrá tener:

- Inteligencia: El XML ofrece la posibilidad de interrelacionar conceptos.
- Adaptabilidad: El XML más que ser adaptable, es usable como lenguaje fuente para el desarrollo de otros lenguajes.
- Mantenimiento: Su mantenimiento es simple, al dividir vínculos de información y de la apariencia.

Por último, aunque el HTML es una arquitectura abierta, arrastra el problema de mover sistemas heredados a nuevos ambientes. El lenguaje XML, fue desarrollado en respuesta

a estas limitaciones del HTML, pues mientras el HTML utilizaba etiquetas predefinidas para describir elementos, el XML permite que esas etiquetas sean definidas por el creador de la página de modo tal que puedan ser ingresadas o extraídas directamente de las bases de datos existentes.

La potencia del XML reside entonces en que permite a los programadores crear sus propias etiquetas para así disponer de la funcionalidad completa del lenguaje de etiquetas. Estas características significa que es extensible, o sea que un programador puede crear libremente sus propias etiquetas.

### 1.7.2. XML y SGML

XML es una versión simplificada de SGML, optimizada para la Web. Al igual que SGML, XML permite determinar un conjunto de etiquetas específico a la hora de describir un documento concreto. También al igual que en SGML, un individuo o un comité de estándares pueden definir una aplicación XML (también denominada vocabulario), que es un conjunto de elementos y una estructura de documento de propósito general, que pueden utilizarse para describir documentos de un determinado tipo. La sintaxis de XML ofrece menos opciones que SGML, haciendo que la lectura de documentos XML sea más sencilla para los humanos, y que la creación de dichos documentos sea más simple para los programadores. <sup>[LIB001]</sup>

### 1.7.3. HTML y SGML <sup>[WWW002]</sup>

El HTML y el SGML se describen como elementos con una relación "padre-hijo". SGML, el lenguaje "padre", es un lenguaje de descripción formal de sistemas de referencia de documentos, es decir, lenguajes "hijo". Aplicando las reglas del SGML pueden crearse cientos de lenguajes "hijo".

El HTML es uno de esos lenguajes "hijo". Mediante instrucciones SGML se define un conjunto de reglas apropiadas para la visualización de documentos en la Web.

Definiéndolo propiamente, HTML es una Definición del tipo de documento (DTD) de SGML. Una DTD es un conjunto de reglas que gobiernan una aplicación específica de SGML, como puede ser el HTML. El HTML DTD define los detalles de la sintaxis del HTML. Ponerse de acuerdo en la sintaxis permite a las compañías de software crear navegadores que visualizarán de igual manera un documento que esté acorde con la sintaxis.

Una característica de SGML es su complejidad, el código SGML esta escrito para ser interpretado por un ordenador, no por una persona. Es difícil leer código SGML y entenderlo a simple vista.

HTML se caracteriza por su manejabilidad, es decir, crear un documento HTML es una tarea asequible. Sin embargo escribir código SGML es un trabajo que puede exceder la capacidad del autor de manejar un documento.