Lenguajes de marcas para la sindicación de contenidos

IES Sant Vicent Ferrer
Algemesí

Contenido

1. Introducción a la sindicación de contenidos

El primer modo de sindicación que hubo fue la integración de noticias de una página dentro de otra por medio de algún tipo de programa que obtenía la información buscando por dentro del contenido HTML. Pero estos programas tenían el problema de que a menudo quedaban obsoletos, ya que cualquier cambio en la página original podía hacer que dejaran de funcionar correctamente.

A pesar de lo que pueda parecer por la abundancia de “decoración” que existe en las páginas web, lo más importante es el contenido. La información contenida en los artículos, los archivos, etc., es lo que hace que los visitantes vuelvan o no.

Con el sistema tradicional de navegación por Internet, para un usuario era muy importante obtener los enlaces de los sitios web que le interesaban y almacenarlos de alguna forma para poder a ellos volver rápidamente. Si se querían seguir los cambios en las páginas web, la única manera que había era ir visitando de vez en cuando la página para comprobar si había novedades.

La aparición de lo que se conoció como Web 2.0 complicó las cosas. La Web se llenó de una gran cantidad de blogs y páginas que publicaban información, y visitarlas todas para ver si había cambios pasó a requerir mucho tiempo. Había que optimizar de alguna manera esta tarea.

La aparición de los sistemas estándar de sindicación hizo posible obtener la información de las actualizaciones de un sitio web de forma estable por medio de una dirección. La sindicación de contenidos cambió la forma en que se recupera la información. Ya no era necesario ir a buscar la información: era la información la que acudía al usuario.

Utilizando la sindicación ya no es necesario que el usuario visite las páginas que le interesan para ver si hay cambios porque si los hay ya los recibirá. Esto comporta un ahorro de tiempo, ya que no será necesario visitar páginas para descubrir que no hay cambios. Con la sindicación el diseño de la página original no afecta a los programas que buscan información, puesto que la sindicación de contenidos está basada en XML y no prioriza la información de estilo sino el contenido.

Ésta está pensada para que puedan interactuar tanto los humanos como los programas, lo que hace que se puedan diseñar fácilmente aplicaciones que obtengan la información de forma automática sin que sea necesario ningún tipo de intervención humana.

La participación del humano se limitará a decir en el programa qué lugares debe vigilar. Además, una vez el usuario recibe la información puede hacer lo que quiera: filtrar sus contenidos, clasificarla por temas... Por tanto, tendrá el control de cuál es la información que quiere ver y cuál no.

Otra de las ventajas que aporta la sindicación es inherente a XML. A diferencia de lo que ocurre con HTML, es fácil interpretar el contenido de la información que se recibe y, por tanto, también será fácil poder reutilizar su contenido para realizar otras tareas.

A pesar de que la sindicación se ve a menudo como un sistema enfocado a detectar novedades en la Web, también se está usando para mantener actualizaciones en otros campos. Por ejemplo, algunos programas de ordenador utilizan RSS para saber si existen actualizaciones nuevas y de esta forma mantienen los programas actualizados.

2. "Feeds" o canales

Un canal es un archivo que contiene una versión específica de la información que se ha publicado en un sitio web.

En este archivo se encuentra toda la información sobre el sitio web y enlaces a sus contenidos.
La gran ventaja es que al estar basado en XML se puede conseguir transmitir la información de forma automatizada y los receptores podrán interpretarla fácilmente.

Para poder obtener la información del canal normalmente será necesario localizar el archivo.
Generalmente estos canales están asociados a una página web y accesible por medio de un enlace.
Los enlaces suelen estar claramente especificados con el texto RSS, XML o haciendo
servir el siguiente grupo de iconos:

Los archivos de los canales normalmente se pasarán a programas que serán los que
se encargarán de recabar periódicamente las actualizaciones de la información del canal.
En la terminología de sindicación esto suele llamarse suscripción.

El uso de canales aportará distintas ventajas a los usuarios:

- Al funcionar por suscripción, éstos sólo recibirán las noticias de interés
  sede.
- El programa tenderá a dar información más detallada a sus preferencias y gustos de lo que lo hacen los buscadores generalistas como Google, ya que los canales contienen el resultado de una búsqueda.
- La información puede clasificarse y ordenarse según los gustos del usuario y consumirse según los criterios de preferencia.
- En cualquier momento se puede dejar de seguir un canal sin tener que pedir
  ningún tipo de permiso.

A lo largo de los años se han desarrollado diversas tecnologías para crear canales como CDF (channel definition format, desarrollado por Microsoft), PointCast o Apple MCF (meta content framework), pero los lenguajes de creación de canales que se han hecho más populares y que se han convertido en el modo estándar de sindicación han sido sobre todo RSS y Atom.

3. RSS

RSS son las siglas que se utilizan para nombrar diferentes estándares muy populares para la sindicación de contenidos que se han convertido en una forma estándar de intercambiar información en la Web.

a. Lenguaje RSS 2.0

Entre todos los canales disponibles, RSS 2.0 es el más usado con mucha diferencia, y por tanto, la gran mayoría de los programas lectores de RSS lo soportan. Una de las características que define RSS es que hace honor a su nombre oficial, really simple syndication (sindicación realmente sencilla), y es un sistema sencillo.

Se trata de un sistema que no tiene ninguna estructura compleja, en el que las etiquetas describen el contenido que hay en el elemento, en el que prácticamente no se utilizan los atributos para nada y los espacios de nombres sólo se utilizan en las extensiones, si las hubiere.

La raíz

RSS es un lenguaje XML, por lo que debe cumplir sus normas y, por tanto, sólo tiene un solo elemento raíz, <rss>. La función de este elemento es simplemente informar a quien lea el documento de que lo que está leyendo es un canal RSS.

El elemento raíz tiene uno de los pocos atributos obligatorios de la especificación, version, que es necesario para indicar la versión que se está utilizando. Este atributo sirve para que los programas sepan qué versión de RSS se utiliza en el documento.

```
<rss version="2.0">
        ...
</rss>
```

El elemento <channel>

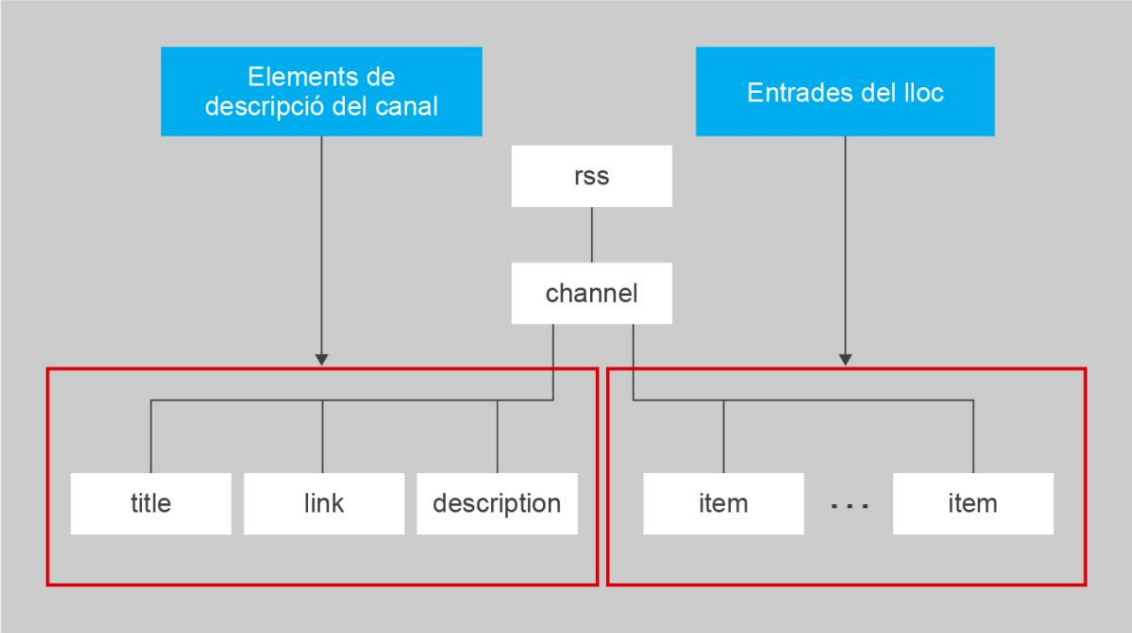
La raíz sólo sirve para indicar que el documento es de tipo RSS, y el contenido del canal estará dentro del único hijo de <rss>, llamado <channel>. El elemento <channel> será el que contendrá todas las etiquetas que aportan información sobre el canal, y sobre todo las que tendrán las novedades del sitio. Sólo puede haber una sola etiqueta <channel> en todo el documento RSS.

```
<rss version="2.0">
    <channel>
    ... contenido ...
    </channel>
</rss>
```

Se puede dividir el contenido de un canal RSS en dos grandes grupos (figura 1):

1. Un grupo de etiquetas destinadas a describir el canal.

2. Los elementos <item>, que son los que contendrán el contenido del canal.



Para hacerlo todo más sencillo, algunas de las etiquetas se repiten en ambos grupos. Por ejemplo, las etiquetas más importantes de <channel>, que son <title>, <link> y <description>, son también las más importantes de los elementos <item>. Etiquetas para describir el canal.

Los primeros elementos que se encuentran dentro del elemento <channel> están destinados a dar información sobre el canal RSS. Estas etiquetas no son de contenido ni será habitual que se produzcan cambios en sus valores.

Los elementos más importantes de esta parte son los elementos <title>, <link> y <description> que son obligatorios en todos los canales RSS.

Estos son los únicos elementos obligatorios y, por tanto, el siguiente documento sería un documento RSS válido.

```
<rss version="2.0">
        <channel>
                <title>Canal de lenguajes de marcas</title>
                <link>http://ioc.xtec.cat/rss/Marques.html</link>
                <description>Canal para entrar las notas del módulo 4 de ASIX</description>

        </channel>
</rss>
```

Aparte de los elementos obligatorios en RSS también hay algunos que son voluntarios y que sirven para dar información extra sobre el canal:

• language: Especifica el idioma en que está escrito el canal. • copyright: Información sobre el copyright del contenido. • managingEditor: Correo electrónico del responsable del contenido. • webMaster: Correo electrónico del responsable técnico. • pubDate: Última fecha de publicación en el canal. • category: Categoría del contenido. • lastBuildDate: Última fecha de modificación del canal.

• generator: Programa utilizado para generar el contenido.
• docs: Describe el formato específico. • ttl: Tiempo que los clientes deben esperar para volver a pedir. • image Icono que representa el canal. • rating Utiliza una clasificación americana sobre el contenido. • cloud Grupo de gente a la que se informa de los cambios. • textInput Es una etiqueta antigua que ya no se utiliza. • skipHours En qué horas no se pueden solicitar actualizaciones. • skipDays Qué días no se pueden solicitar actualizaciones.

Contenido del canal

El contenido visible de un canal RSS irá en los elementos <item>, de los que puede haber la cantidad que se desee (incluso ninguna).

Normalmente cada vez que se produce una novedad en el sitio asociado al contenido de un canal se crea un nuevo elemento item que se añade al documento.

A pesar de que no parece tener sentido crear ítems sin contenido, esto es estrictamente posible porque ítem no tiene ningún elemento obligatorio pero siempre debe haber un <description> o un <title>.

Por tanto, lo siguiente sería un documento RSS correcto. Tenemos dos ítems, uno con título sin contenido y uno con contenido sin título.

```
<rss version="2.0">
        <channel>
                <title>RSS</title>
                        <link>http://ioc.xtec.cat/rss/RSS</link>
                        <description>Provant</description> <item>

                        <title>Título sin contenido</title> </item> <item>


                        <description>Contenido sin título</description> </item> </
                channel>
        </rss>
```

4. Atom

Atom fue diseñado pensando en superar los problemas de interpretación que tenía RSS 2.0 y evitar la complejidad añadida de RSS 1.0. Su idea era aprovechar las mejores cosas de los RSS y arreglar las partes que causaban confusión.

Un segundo objetivo que le diferencia claramente de RSS es que también se quería que no sólo servirá para recuperar los cambios en la información del canal sino que también se pudiera utilizar de forma estandarizada para añadir información. La sindicación ha estado muy ligada a los blogs y hasta el momento cada programa para hacer blogs utilizaba su protocolo propio (Blogger API, MetaWebLog API, ...), que estaban pensados sólo para un blog en concreto.

Por tanto, podemos dividir Atom en dos partes:

- Atom syndication format: un lenguaje XML para sindicar contenidos.
- Atom publishing protocol: un protocolo basado en HTTP pensado para actualizar y crear recursos en la Web.

Aparte de las diferencias en las etiquetas, las grandes diferencias con RSS son que:

- Permite definir cuál es el contenido de las etiquetas (texto, HTML, etc.), pero también permite referencias a archivos externos. En RSS no se define qué contenido existe.
- Se puede utilizar dentro de otros documentos XML, ya que tiene su definición y utiliza los espacios de nombres. No se puede poner RSS dentro de otros documentos XML porque no tiene en cuenta el espacio de nombres.
- RSS no ofrece un protocolo de publicación como Atom.

a. Lenguaje Atom

Atom sólo tiene una raíz, que es <feed>. Esta etiqueta la pueden utilizar los programas para detectar que el documento que están leyendo es de tipo Atom.

La raíz <feed> siempre debe tener definido el espacio de nombres de los documentos Atom, que es www.w3.org/2005/Atom. Si no se especifica el espacio de nombres, el documento no validará.

```
<feed xmlns="http://www.w3.org/2005/Atom">
        ...
</feed>
```

El hecho de tener un espacio de nombres y utilizarlo posibilita que los documentos Atom se puedan mezclar con documentos XML de otros vocabularios sin problemas.

Atom tiene disponibles los atributos de XML xml:lang, que sirve para identificar el idioma del documento, y xml:base, que se utiliza para controlar cómo se resuelven las direcciones relativas.

Al igual que en otros lenguajes de canales, como RSS, se pueden agrupar las etiquetas de Atom en dos grupos:

- Etiquetas que proporcionan datos sobre el canal.
- Etiquetas con el contenido del canal.

Etiquetas con datos del canal

Los elementos obligatorios dentro de la etiqueta <feed> siempre deben tener los elementos hijos <title>, <id> y <updated>

Por tanto, éste sería un documento Atom válido, ya que éstas son las únicas etiquetas obligatorias:

```
<?xml version="1.0" encoding="utf 8"?>
<feed xmlns="http://www.w3.org/2005/Atom">
        <title>Atom IOC</title>
        <updated>2011-08-13T15:20:02Z</updated>
        <id>http://ioc.xtec.cat/</id>
</feed>
```

Aparte de los elementos obligatorios, en la especificación de Atom también se hace referencia a unos elementos llamados “muy recomendables” Los elementos recomendables son <link> y <author>.

Etiquetas de contenido del canal

Para definir las diferentes entradas dentro de un canal se utiliza como base el elemento <entry>. Cada nueva aportación creará un nuevo elemento <entry>, que al menos debe tener las etiquetas <title>, <id>, <updated>, y además un elemento <content> o un elemento <link>

Como hace a menudo la especificación, Atom también define un segundo nivel de elementos, considerados “muy recomendados”, y que, por tanto, también deberían salir. Entre los recomendados habrá <content> o <link> si no se han especificado anteriormente.

El elemento <author> se convierte en obligatorio si no se ha especificado ninguno en los metadatos del canal.

Formato de las fechas

Atom utiliza el RFC 3339 (ISO 8601) para definir el formato de las fechas. Las fechas en Atom deben tener esta forma:

Año-Mes-DíaTHora:Minutos:Según-zonahoraria

De modo que:

- Todos los valores son numéricos excepto la zona horaria, que en algunos casos puede ser el carácter "Z" para indicar la hora universal.
- Delante de la zona horaria se especifican las horas de retraso o de adelanto con los símbolos de suma o resta.
- Se utiliza la letra “T” para separar los días de las horas.

El contenido

Si no se especifica ningún atributo en el elemento <content> o en <summary>, éste será tratado como si fuera texto plano. Si se quiere dejar claro que el contenido está en algún otro formato debe especificarse con el atributo type, que normalmente tendrá los valores “texto”, “html” o “xhtml”.

5. Agregadores / lectores

Los agregadores y lectores de feeds son programas que permiten al usuario mantener en un solo sitio toda la información de los canales que le interesan.

Entre otras cosas:

- Se encargan de actualizar los cambios que se van produciendo sin que el usuario tenga que visitar la página.
- Llevan el control de los contenidos leídos y no leídos de los canales.
- Permiten ver un resumen de las noticias de un sitio web.
- Se pueden organizar las noticias en grupos personalizados.
- Permiten realizar búsquedas de información entre la información del canal.

Tanto RSS como Atom son estándares abiertos, lo que ha permitido que se hayan creado una gran cantidad de lectores que los soportan y que ofrecen funciones extra para los usuarios para intentar mejorar su experiencia.

A pesar de que se pueden leer los canales desde diferentes programas de correo o los navegadores, normalmente se consigue trabajar de forma más cómoda y personalizable utilizando programas especializados.

Por lo general podemos dividir los programas lectores de canales en dos grandes grupos:

- Lectores web
- Lectores de escritorio

A pesar de la división normalmente el aspecto visual de estos programas es relativamente similar. Todos suelen tener la pantalla dividida en bloques:

- En uno de los blogs suele haber la lista de suscripciones en la que se pueden agrupar estas suscripciones por temas.
- Un blog para mostrar el contenido de cada una de las suscripciones.

Ejemplo de agregadores populares

Lectores vía web	Software de escritorio
Feedly	Liferea
Netvibes	SharpReader
BlogLines	FeedDemon
FeedLooks	FeedReader

6. Bibliografía y referencias

Sala, Javier. (2023) Lenguajes de marcas y sistemas de gestión de información.

Instituto Abierto de Cataluña