



# Aplicación de los lenguajes de marcas a la sindicación de contenidos

Lenguaje de Marcas y  
Sistemas de Gestión de  
la Información



# Índice



- 7.1. Ventajas y ámbitos de aplicación**
- 7.2. Tecnologías y estándares de canales de contenidos**
  - 7.2.1. RSS
  - 7.2.2. Atom
- 7.3. Estructura y elementos de los canales de contenidos**
  - 7.3.1. Estructura y elementos de RSS 2.0
  - 7.3.2. Estructura y elementos de Atom
- 7.4. Creación, validación y comprobación de funcionalidades de los canales de contenidos**
  - 7.4.1. Creación de un feed RSS
  - 7.4.2. Validación y comprobación
- 7.5. Utilización de herramientas**
  - 7.5.1. Herramientas de creación y edición de feeds RSS
  - 7.5.2. Agregadores o lectores de feeds
- 7.6. Directorios de canales de contenidos**



## Introducción

Vivimos en un mundo donde la globalización y la web 2.0 han hecho que la información sea una mercancía extremadamente abundante, tanto que se entrega en una cantidad ingente que como personas no somos capaces de procesar. Con el fin de poder estar informados es imprescindible cribar dicha información para poder alcanzar aquella que realmente nos interesa, pero si tuviéramos que descubrir el contenido de cada noticia sería imposible, hay es donde la sindicación de contenidos muestra su valor.

La sindicación permite la creación de feed, pequeños extractos de noticia, con título y breve introducción, que nos permiten navegar rápidamente entre las distintas noticias hasta alcanzar aquellas

que nos interesan. Aún con esto puede resultar una tarea tediosa, por lo que usando este mismo sistema se han creado numerosas herramientas como los agregadores para facilitar esta tarea.

Los agregadores permiten crear un perfil del usuario y la información que desea para después filtrar la información y ofrecérsela al usuario, con solo las noticias relevantes para él y actualizadas.

En este tema veremos las dos caras de la misma moneda, cómo crear feeds de noticias y cómo automatizar su recepción para que tan solo nos llegue lo que realmente nos interesa.

## Al finalizar esta unidad

- + Estudiaremos los lenguajes de marcas usados en la sindicación de contenidos.
- + Conoceremos las distintas herramientas existentes para la creación automática de feeds, su edición y validación.
- + Describiremos cual es la estructura y sintaxis de los dos lenguajes de marcas usados en la sindicación de contenidos como son RSS y Atom.
- + Descubriremos los distintos medios y herramientas con los que podemos recibir y filtrar los feeds publicados.

# 7.1.

## Ventajas y ámbitos de aplicación

En un mundo como el nuestro que avanza tan rápidamente, la información queda desactualizada con gran en ocasiones en horas, lo que supone para las personas que buscan estar el día un gran problema a la hora de buscar información o más bien lo harían, si no existiera la sindicación de los contenidos, la cual nos permite encontrar en una única página web la información más nueva siempre primero.



Esta sindicación actúa de la misma manera que nuestra bandeja de entrada de correo, manteniendo siempre lo más nuevo e importante siempre a la vista.

Es, en definitiva, el acto de compartir publicaciones de otras páginas en una sola, de modo que visitando una única página podemos enterarnos de todas las noticias actuales y, si alguna nos interesara, visitar la página de la que proviene a través de un enlace.

Esta manera de organización, entre otras ventajas que se comentarán a continuación, la principal razón del uso de este tipo de organización es el intento de facilitar a los usuarios a la información más actual y relevante.

Las ventajas más relevantes de la sindicación de los contenidos son:

- > La actualización periódica automática ahorra una gran cantidad de tiempo de búsqueda a los usuarios.
- > Es más preciso que el uso de buscadores.
- > Acelera la búsqueda de noticias.
- > Permite el filtrado, personalizando así los temas de los contenidos recibidos.
- > Evita el Spam, la publicidad excesiva y posibles virus a los que nos expondríamos buscando en diferentes páginas a través de buscadores.
- > Permite acceder a noticias en tiempo real.
- > La subscripción en una única página.
- > Etc.

La RSS, Really Simple Syndication, utilizan el texto plano para distribuir las noticias en la red con un formato XML.



## 7.2.

### Tecnologías y estándares de canales de contenidos

La sindicación usa distintos formatos, pero entre ellos destacan RSS y ATOM, los cuales estudiaremos a continuación. Ambos están basados en el lenguaje XML, por lo que son lenguajes de marcas, y se usan para la difusión rápida de información, principalmente en el ámbito mediático.

Se suele emplear un icono de RSS o ATOM por parte de la propia página para marcar que usan un sistema de sindicación de contenido, ya sea uno genérico, o especificando el formato usado. Este icono posee siempre un color naranja.

Es común que las páginas que usan este elemento utilicen también un feed, conocido como fuente web o canal. Este feed es un XML con la información necesaria, que permite a los usuarios visitar la página de la que proviene la información.



Imagen 1. Iconos de sindicación de contenidos.

#### 7.2.1. RSS

Las RSS, Really Simple Syndication o sindicalización realmente simple, es el más usado y, por tanto, el que marca el estándar al resto de formatos usados. Es un lenguaje proveniente del XML por lo que es considerablemente sencillo de manejar para la escritura de texto plano, en especial si se posee una plantilla preprogramada, ahorrando mucho tiempo y agilizando el proceso.

El formato RSS ha ido evolucionando a lo largo de los años, llegando a cambiar de nombre.

Dave Winer, un creador de software, crea RSS en el año 1997.

Dos años después Netscape, con las siglas significando rich site summary, resumen óptimo del sitio, crea las versiones RSS 0.90 y RSS 0.91.

Un año después en el 2000, Dave Winer, lanza RSS 0.92 con nuevas características. Ese mismo año O'Reilly desarrolla RSS 1.0, el cual significaba RDF site summary, ya que el formato utilizado era RDF.

En 2002, una vez más Dave Winer, crea RSS 2.0, Really Simple Syndicate, es el que se sigue empleando hoy en día.



Imagen 2. Icono de RSS.

#### 7.2.2. Atom

Es un formato muy similar al RSS, ya que ambos proceden de XML y tienen el mismo uso. Fue creado con la intención de arreglar fallos de RSS y simplificarlo. Fue creado en 2005 por Internet Engineering Task Force, IETF, pero a pesar de considerarse mejor, ya que soluciona fallos y aumenta las características de RSS, sigue siendo menos utilizado.



# 7.3.

## Estructura y elementos de los canales de contenidos

Como un derivado de XML, requiere de una sintaxis rígida con unas normas muy claras. Además, dado que tiene una función específica, cuenta con determinados elementos obligatorios. Casi podemos decir que es un simple XML con una plantilla específica.

### 7.3.1. Estructura y elementos de RSS 2.0

Los elementos que destacan en la estructura de los ficheros RSS, e importantísimos para una noticia, son los siguientes:

- > **Título.** Obligatorio.
- > **Resumen.** Obligatorio.
- > **Enlace o hipervínculo a la noticia completa.** Obligatorio.
- > **Autor.** Opcional.

La estructura básica y obligatoria de un RSS es la siguiente:

```
<?xml version="1.0" encoding="utf-8"?>
<rss version="2.0">
  <channel>
    <title></title>
    <link></link>
    <description></description>
  </channel>
</rss>
```

Como XML debe comenzar siempre, como ya hemos estudiado, con la línea introductoria que marca su versión.

```
<?xml version="1.0" encoding="utf-8"?>
```

La segunda línea marca el archivo como un fichero RSS, marcando la versión utilizada.

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

La siguiente es la raíz, sus elementos son los que darán la información relevante, existen tres elementos obligatorios y 16 opcionales.

```
<channel>
```





## Los elementos requeridos de <channel>

Existen tres elementos obligatorios en este apartado:

1. **<title>** Marca el título de la noticia o anuncio.
2. **<link>** Contiene un enlace o hipervínculo que nos redirige con una URL estructurada como una URI, uniform resource identifier.
3. **<description>** Introduce el recurso con un breve resumen.

```
<?xml version="1.0" encoding="utf-8"?>
```

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

```
  <channel>
```

```
    <title>Identificador de recursos uniforme</title>
```

```
    <link>https://es.wikipedia.org/wiki/Identificador_de_recursos_uniforme</link>
```

```
    <description>Esta página web nos introducirá en el concepto de RLI</description>
```

```
  </channel>
```

```
</rss>
```

## Elementos secundarios de <channel>

Existen un total de 16 elementos secundarios u opcionales, los cuales añaden información a los feed. Estos elementos son los siguientes:

Elementos secundarios de <channel>					
managingEditor	language	webMaster	lastbuildDate	generator	copyright
docs	ttl	image	rating	cloud	textInput
skipHours	skipDays	pubDate		category	

- > **<category>** Con el atributo opcional "domain" incluye diversas categorías del feed.

```
<category domain=" " > plantas/flores </category>
```

- > **<docs>** La URL con el documento que contiene el formato del feed.

```
<docs>http://blogs.law.harvard.edu/tech/rss </docs>
```

- > **<ttl>** Marca el tiempo, en minutos que permanecerá en el caché antes de ser actualizado.

```
<ttl>60</ttl>
```

- > **<pubDate>** En formato RFC 822, señala la última fecha de publicación.

```
<pubDate>Wed, 16 Mar 2022 12:16:10 +0100</pubDate>
```



- > **<Image>** Elemento GIF, PNG o JPEG, que representa el feed, contiene tres subelementos obligatorios y otros tres opcionales.

» **Obligatorios:**

- + **<url>** URL del elemento.
- + **<title>** título de la imagen o descripción.
- + **<link>** Enlace al sitio web.

» **Opcionales:**

- + **<description>** En el atributo de título HTML del enlace de la imagen, descripción que precisa el texto.
- + **<height>** Alto de la imagen en píxeles, cuenta por defecto con el valor de 31, hasta un valor máximo de 400.
- + **<width>** Ancho de la imagen en píxeles, cuenta por defecto con el valor de 88, hasta un máximo de 144.

```
<image>
  <url>http://www.butanic.com/imagenes/rosa.jpg</url>
  <title>Rosa en flor</title>
  <link> http://www.butanic.com</link>
  <description>florecimiento de las flores</description>
  <height>200</height>
  <width>88</width>
</image>
```

- > **<copyright>** Derechos de autor del contenido de la publicación.

```
<copyright>Copyright 2022 butanic</copyright>
```

- > **<lastBuildDate>** En formato RFC 822, la fecha y hora de la última modificación.

```
<lastBuildDate>Wed, 16 Mar 2022 12:16:10 +0100</lastBuildDate>
```

- > **<rating>** Clasificación PICS, platform for internet content selection, del feed, Puedes encontrar mas información sobre los PICS en la página <https://www.w3.org/PICS/>.

- > **<skipDays>** Pudiendo introducir como subelementos hasta los siete días de la semana, con <day>, determina los días en los que no se debe actualizar el feed.

```
<skipDays>
  <day>Monday</day>
  <day>Saturday</day>
  <day>Sunday</day>
</skipDays>
```





- > **<skipHours>** Pudiendo introducir como subelementos hasta las 24 horas del día, con <hour>, determina los días en los que no se debe actualizar el feed. Las horas se numeran desde el 0 al 23.

```
<skipHours>
  <hour>0</hour>
  <hour>1</hour>
  <hour>2</hour>
  <hour>3</hour>
  <hour>4</hour>
</skipHours>
```

- > **<generator>** Indica el programa que ha creado el fichero RSS.
- > **<language>** Indica el lenguaje del feed. Se debe usar un código aprobado, puedes encontrarlos en esta página:

[https://www.w3schools.com/tags/ref\\_language\\_codes.asp](https://www.w3schools.com/tags/ref_language_codes.asp)

```
<language>es</language>
```

- > **<cloud>** Para registrar, mediante atributos, procesos en la nube y recibir notificaciones de las actualizaciones del feed. Es necesario que el servicio web especificado admita la interfaz rssCloud y sea capaz de implementar HTTP-POST, XML-RPC o SOAP 1.1.

```
<cloud domain="" port="" path="" registerPro-
cedure="" protocol="" />
```

- > **<managingEditor>** Informa del correo del editor, para posibles consultas.

```
<managingEditor> 12345@Editors.com </managingEditor>
```

- > **<webMaster>** Informa del correo del webmaster o responsable de la web, para posibles consultas.

```
<webMaster> 00002@Editor.com </webMaster>
```

- > **<textInput>** Define un campo de texto de entrada con cuatro subelementos obligatorios, este elemento es poco usado.

- » **<description>** Descripción de la entrada de texto con un máximo de 500 caracteres.

- » **<link>** URL del script CGI el cual se encarga de procesar el ingreso de texto.

- » **<title>** Determina la etiqueta del botón enviar con un máximo de 100 caracteres.

- » **<name>** Indica el nombre del objeto de texto.



## Elementos de <item>

Los elementos <item> guardan información relacionada con el feed, aumentando su valor informativo, y manteniéndolo actualizado y contextualizado, aunque es muy común no es obligatorio.

La información en él es visible en el feed, y consta de diez elementos opcionales, aunque es obligatorio que uno de estos dos <title> o <description> aparezca.

Subelementos de <item>				
title	link	description	source	guid
pubDate	author	category	comments	enclosure

- > **<title>**: Título del ítem.
- > **<link>** Contiene un enlace o hipervínculo que nos redirige con una URL.
- > **<guid>** Identificador global único, puede contener un atributo «isPermaLink="» cuyo valor será "true" si el código es una URL, en caso contrario será "false".

`<guid isPermaLink=" " > </guid>`

- > **<enclosure>** Es usado para añadir archivos multimedia. Contiene tres atributos.

- » **url**: dirección URL del elemento.
- » **length**: tamaño en bytes.
- » **type**: tipo de archivo.

`<enclosure url=" " length=" " type=" " />`

- > **<author>** Dirección de correo del autor del artículo para posibles consultas.
- > **<source>** Título del RSS del que proviene el artículo. Contiene de manera obligatoria el atributo "url", con la dirección de dicho artículo.

`<source url=" " > </source>`

- > **<comments>** Especifica una URL donde los lectores pueden dejar comentarios.
- > **<pubDate>** En formato RFC 822, la fecha y hora de publicación.
- > **<category>** Determina la categoría, pueden aparecer varias.
- > **<description>** Un resumen del artículo o incluso algunas frases extraídas de él. Se escribe en texto plano o usando un formato HTML.



Un ejemplo de la estructura, vacía, de un feed con ítems es este.

```
<?xml version="1.0" encoding="utf-8"?>
<rss version="2.0">
  <channel>
    <title></title>
    <link></link>
    <description></description>
    <item>
      <title></title>
      <link></link>
      <description></description>
    </item>
    <item>
      <title></title>
      <link></link>
      <description></description>
      <author></author>
    </item>
  </channel>
</rss>
```



### 7.3.2. Estructura y elementos de Atom

La estructura de Atom, en su versión actual ATOM 1.0, es similar a la de RSS.

```
<?xml version="1.0" encoding="utf-8"?>
<feed xmlns="http://www.w3.org/2005/Atom">
  <title>
</title>
  <entry>
</entry>
  <entry>
</entry>
</feed>
```

Como XML sigue manteniendo la primera línea, pero la segunda se cambia para reflejar que es un elemento Atom mediante la siguiente frase:

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

Es seguido por la raíz <feed> la cual, obligatoriamente, contiene el atributo "xmlns" la cual define el tipo de documento:

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

#### Elementos de <feed> requeridos, recomendables y opcionales

Todos los elementos de <feed> que posean texto pueden contener el atributo "type" el cual marca el tipo de texto con los valores "text", "html" y "xhtml".

Los elementos de <feed> son los siguientes:

- > **<id>** Muestra la URI del canal.
- > **<title>** El nombre del canal, es obligatorio.
- > **<updated>** Marca la última modificación en formato RFC 3339. La fecha y hora se separan con una "T" mayúscula y la "Z" final nos indica que nos guiamos por la hora universal, en caso contrario se especifica la zona horaria.  

```
<updated>2022-03- 18T08:16:12Z</updated>
```
- > **<link>** Páginas web relacionadas. Posee un atributo obligatorio y varios opcionales.
  - » **href:** Obligatorio, URI del recurso empleado.
  - » **rel:** URI con uno de los siguientes valores:
    - + **alternate.** El usado por defecto.
    - + **enclosure.**
    - + **related.**
    - + **self.**
    - + **via.**



- » **type**: El tipo de recurso empleado.
  - » **hreflang**: Idioma del recurso empleado con un código, la lista de códigos se encuentra en: [https://www.w3schools.com/tags/ref\\_language\\_codes.asp](https://www.w3schools.com/tags/ref_language_codes.asp)
  - » **title**: Título del recurso.
  - » **length**: En bytes, el tamaño del recurso.
- ```
<link href=" " type=" " hreflang="es" title=" "/>
```
- > **<author>** autor del canal, puede contener tres subelementos, uno obligatorio y dos opcionales.
    - » **<name>** Obligatorio, Identifica el nombre del autor.
    - » **<email>** Correo del autor para posibles consultas.
    - » **<uri>** URI de la página web del autor.

```
< author>  
  <name></name>  
  <email></email>  
  <uri></uri>  
</ author>
```
  - > **<subtitle>** Determina el subtítulo y su creación posee la misma estructura que <title> con su atributo opcional "type", por ser un elemento con texto, y sus valores "text", "html" y "xhtml".
  - > **<rights>** Contiene los derechos de autor del canal.
  - > **<contributor>** Identifica a un colaborador del canal, cuenta con la misma estructura que el elemento <author>.
    - » **<name>** Obligatorio, Identifica el nombre del autor.
    - » **<email>** Correo del autor para posibles consultas.
    - » **<uri>** URI de la página web del autor.

```
<contributor>  
  <name></name>  
  <email></email>  
  <uri></uri>  
</contributor>
```
  - > **<generator>** Software usado para la creación del canal, cuenta con dos atributos opcionales: "uri" y "version".

```
<generator> uri=" " versión=" "></generator>
```
  - > **<logo>** Dirección de una imagen insertada.
  - > **<icon>** Dirección de una imagen insertada.
  - > **<category>** Especifica la categoría del canal, cuenta con tres atributos uno de ellos obligatorio:
    - » **term**: obligatorio, en él se especifica la categoría del elemento.
    - » **scheme**
    - » **label**

```
<category term=" "/>
```



## Elementos de <entry> requeridos, recomendados y opcionales

El feed, puede contener una o varias entradas <entry>, estos elementos constan de los siguientes subelementos:

- > **<id>** Identificador URI.
- > **<title>** Obligatorio, Marca el título del elemento.
- > **<updated>** Última fecha de modificación en formato RFC 3339. La fecha y hora se separan con una "T" mayúscula y la "Z" final nos indica que nos guiamos por la hora universal, en caso contrario se especifica la zona horaria.  
`<updated>2022-03- 18T08:16:12Z</updated>`
- > **<content>** Muestra el contenido, escribiéndolo o con un enlace, es obligatorio si no hay elemento <summary> u otro enlace alternativo que muestre el contenido.
- > **<summary>** Resumen o algunas líneas del contenido de la publicación.
- > **<rights>** Contiene los derechos de autor del canal.
- > **<contributor>** Identifica a un colaborador del canal, cuenta con la misma estructura que el elemento <author>.
  - » **<name>** Obligatorio, Identifica el nombre del autor.
  - » **<email>** Correo del autor para posibles consultas.
  - » **<uri>** URI de la página web del autor.

`<contributor>`

`<name></name>`

`<email></email>`

`<uri></uri>`

`</contributor>`

- > **<published>** Fecha de publicación en formato RFC 3339. La fecha y hora se separan con una "T" mayúscula y la "Z" final nos indica que nos guiamos por la hora universal, en caso contrario se especifica la zona horaria.

`<published>2022-03- 18T08:16:12+0100</published>`

- > **<source>** Especifica los datos de otros elementos si el elemento <entry> es una copia.

`<source>`

`<id><id>`

`<title></title>`

`<updated></updated>`

`<rights></rights>`

`</source>`



> **<author>** autor del canal, puede contener tres subelementos, uno obligatorio y dos opcionales.

- » **<name>** Obligatorio, Identifica el nombre del autor.
- » **<email>** Correo del autor para posibles consultas.
- » **<uri>** URI de la página web del autor.

< author>

<name></name>

<email></email>

<uri></uri>

</ author>

> **<link>** Páginas web relacionadas. Posee un atributo obligatorio y varios opcionales.

- » **href**: Obligatorio, URI del recurso empleado.
- » **rel**: URI con uno de los siguientes valores:
  - + **alternate**. El usado por defecto.
  - + **enclosure**.
  - + **related**.
  - + **self**.
  - + **via**.
- » **type**: El tipo de recurso empleado.
- » **hreflang**: Idioma del recurso empleado con un código, la lista de códigos se encuentra en: [https://www.w3schools.com/tags/ref\\_language\\_codes.asp](https://www.w3schools.com/tags/ref_language_codes.asp)
- » **title**: Título del recurso.
- » **length**: En bytes, el tamaño del recurso.

<link href=" " type=" " hreflang="es" title=" "/>

> **<category>** Especifica la categoría del canal, cuenta con tres atributos uno de ellos obligatorio:

- » **term**: obligatorio, en él se especifica la categoría del elemento.
- » **scheme**
- » **label**

<category term=" "/>



# 7.4.

## Creación, validación y comprobación de funcionalidades de los canales de contenidos

Es común que los ficheros RSS se creen de manera automática usando otro tipo de lenguaje, en especial en lugares como blog, por lo que es recomendable revisar esos ficheros en busca de errores producidos por la incompatibilidad o durante el proceso de traducción.

### 7.4.1. Creación de un feed RSS

La manera más sencilla de crear un feed es el uso de una herramienta que lo genere automáticamente, por lo que solo deberemos rellenar las casillas que el programa nos indique con la información deseada y la herramienta traducirá esos datos en un fichero RSS funcional, lo que nos ahorrará mucho tiempo y evitará errores.

Otra forma, mucho más laboriosa, es crearlo desde cero usando programas de escritura XML, como es Notepad++, donde deberemos crear la estructura y añadir el contenido. Esta última nos da mucha más libertad al costo de un mayor trabajo y riesgo de errores, aunque esto último se puede paliar si prefabricamos una estructura y la usamos de "molde" para nuestros ficheros RSS.

Los pasos que debemos seguir para crear un fichero RSS de forma manual son los siguientes:

1. Tener clara la estructura que debemos elaborar en función de la información y referencias que vayamos a añadir, como, por ejemplo, el número de <item>.
2. Crear el documento XML y añadir las líneas iniciales que lo marquen como un fichero con formato RSS, incluyendo su versión y demás elementos obligatorios. Si disponemos de ella es aconsejable usar una plantilla, ya que ahorraremos tiempo y evitaremos errores.
3. Añadir los datos del feed comprobando que todos los campos obligatorios se hayan rellenado.
4. Insertar los elementos opcionales y subelementos deseados.
5. Guardar el archivo como un fichero con extensión .rss o, en su defecto, .xml.
6. Validar el feed, tal como veremos en el siguiente punto.
7. Publicar el feed con FTP o un gestor de archivos.





### 7.4.2. Validación y comprobación

Como todo elemento creado por manos humanas existe la posibilidad de que se hayan cometido errores durante su fabricación. Estos errores pueden no ocasionarnos problemas, pero también pueden provocar que el archivo no se cargue bien o no pueda ser leído directamente, por lo que es muy importante revisar nuestro trabajo en busca de los posibles fallos cometidos.

Con el fin de llevar a cabo esta tarea existen diversos métodos.

#### Servicio de validación online del W3C

A través de URI o con una estrada directa, este servicio gratuito nos permite verificar la sintaxis de los archivos RSS y Atom.

Esta validación no corregirá los errores si los hay, pero nos señalará dónde está, a través del número de línea, y que tipo de errores.

Además de los errores, la aplicación nos puede dar consejos sobre ciertos asuntos si lo ve oportuno, basándose en nuestros errores.

#### Feed Validator

Otra herramienta es Feed Validator, que al igual que la herramienta de W3C, nos permite comprobar los errores en RSS, Atom y RDF Site Summary, marcando los errores para que podamos corregirlos nosotros mismos.

#### Herramienta software FeedForAll

Esta herramienta de creación de feed incorpora su propio validador, lo que nos permite crear y corregir con la misma herramienta.

Puedes encontrar esta herramienta en su página digital: <https://www.feedforall.com/>



Imagen 3. La validación es un proceso importante para evitar que nuestro trabajo sea en vano por culpa de un fallo.

# 7.5.

## Utilización de herramientas

Debido a la gran difusión de los ficheros RSS la demanda de su creación se disparó, lo que a su vez provoca que se creen diversas herramientas para su creación.

Debemos tener en cuenta que la mayoría de los ficheros RSS son creados por personas que no saben cómo crear o escribir un archivo en formato XML, por lo que tampoco son capaces de trabajar con ficheros RSS o Atom.

La solución a este problema es la creación de herramientas que faciliten la creación de esos feeds, un intermediario que convierta las ideas de los usuarios en feed de manera sencilla. Estas herramientas permiten que los usuarios puedan crear sus feeds, rellenando únicamente diferentes casillas de texto que después la propia herramienta se encargará de traducir a lenguaje RSS o Atom de manera automática, permitiendo que los usuarios se centren únicamente en el contenido y no en la creación en sí.



Imagen 4. La difusión de este tipo de fichero, al tratar con contenido mediático se extiende por todo el mundo, y gracias a los blogs es cada día más usado.



### 7.5.1. Herramientas de creación y edición de feeds RSS

---

Existen numerosas herramientas para la creación de ficheros RSS, algunas de las más conocidas son las siguientes:

- > FeedForAll
- > RSS Builder
- > ListGarden
- > Absolute RSS Editor
- > RSS Feed Creator

A continuación, se dará una breve explicación de cada una de ellas.

#### FeedForAll

Una herramienta bastante completa que nos permite crear, editar, validar y publicar en formato RSS.

Su uso es bastante sencillo debido a su asistente que nos guiará por los campos que debemos o podemos rellenar.

Podemos encontrarlo en: <https://www.feedforall.com/>

#### RSS Builder

Una herramienta freeware para la creación de ficheros RSS.

Es muy sencillo de usar, tan solo se deben rellenar los campos indicados, incorporando, para el campo <description> un editor HTML.

Permite la publicación a través de un cliente FTP.

Podemos encontrarlo en: <https://rss-builder.softonic.com/?ex=CAT-3508.5>

#### ListGarden

Programa de código abierto para la creación de feed RSS y podcasts.

Se ejecuta sobre el navegador web, no sobre el escritorio.

De fácil uso permite el almacenamiento local o remoto con un FTP.

Podemos encontrarlo en: <http://softwaregarden.com/products/listgarden/>

#### Absolute RSS Editor

Herramienta diseñada para crear y editar RSS 2.0 y podcasts para iTunes.

Se rellena mediante campos de texto.

Incorpora un validador, y permite una vista previa y la publicación.

Podemos encontrarlo en: <https://www.alentum.com/rsseditor/>

#### RSS Feed Creator

Permite crear y editar ficheros RSS y podcasts.

Un diseño sencillo e intuitivo.

Podemos encontrarlo en: <https://rss-feed-creator.softonic.com/?ex=CAT-3508.5>



### 7.5.2. Agregadores o lectores de feeds

Este tipo de software permite al usuario, previa suscripción, leer y administrar las noticias que un blog, medio de comunicación o cualquier otra página web que use feed contiene. Esta herramienta permite seleccionar dichas noticias y filtrarlas con las preferencias personales del usuario, como por ejemplo solo noticias de deporte, de modo que al usuario tan solo lleguen los artículos deseados, sin que tenga que buscarlos entre el resto de los artículos.

Los navegadores suelen contener este tipo de herramienta para poder personalizar la experiencia del usuario, pero existen diversas herramientas de escritorio diseñadas para cumplir esta misma función.

Es, en definitiva, un paso intermedio entre los artículos y el usuario que le ayuda a filtrar las noticias no deseadas y eliminando las ya leídas o con contenido repetido, evitando que este se abrume por la sobre información.

Algunos de los agregadores son:

#### Agregadores online o web

Cuentan con la ventaja de permitir el uso de esta herramienta sin ningún tipo de instalación y, vinculándola a una cuenta, usarla indistintamente en diversos equipos.

Alguno de los más usados y conocidos son los siguientes.

- > **Feedly:** disponible también para móviles permite la gestión actualizada y personalizada de feeds. Requiere un registro con una cuenta de cuenta, y nos permite visualizar la información de diversas páginas en varios formatos a nuestra elección.
- > **Inoreader:** disponible también para móviles. Requiere un registro con una cuenta de cuenta, y nos permite etiquetar diversas páginas con el fin de organizar de manera automática los artículos. Podemos poseer cuentas sin límites gratuitamente.
- > **Digg Reader:** disponible también para móviles. Permite dos modos de lectura y se caracteriza por su sencillez. Requiere registro.
- > **The Old Reader:** Fácil de usar y gratuito por debajo de los 100 feeds. Requiere de registro, y pago en caso de cuenta premium, cuenta con una app para la mayoría de las gestiones importantes.



## Agregadores de escritorio

Requieren de instalación y aunque algunos de ellos poseen aplicaciones para móviles y tabletas, no todos las poseen. Suelen contar con una interfaz intuitiva y generalmente conocida, como es la del correo electrónico.

Algunas de estas aplicaciones son:

- > **Omea Reader:** Es gratuito y actúa como agregador, gestor de tareas, agenda, etc. Es de fácil uso y se recomienda por su clasificación y organización.
- > **Liferea:** su nombre completo es Linux Feed Reader, software libre con licencia GNU de Linux. Soporta podcasts y se adapta a GNOME.
- > **FeedReader:** agregador que dispone tanto de versión de escritorio como online.
- > **FeedDemon:** agregador para RSS y ATOM en Windows, con una interfaz "amigable" permite la filtración de feed más antiguos.
- > **NewzCrawler:** agregador para RSS y ATOM en Windows.
- > **RSSReader:** agregador freeware para RSS y ATOM en Windows.
- > **NewsFire:** agregador RSS para Mac Os X.

## Agregadores en navegador web

Vienen ya incluidos en el propio navegador y cuentan con las mismas ventajas que los Agregadores online o web.

Los más relevantes son:

- > **Mozilla Firefox:** permite la subscripción con solo pulsar el icono de feed, por lo que su uso es extremadamente sencillo.
- > **Opera:** Permite recibir como si de correos electrónicos se tratase las últimas noticias de los sitios en los que nos hemos suscrito.

## Agregadores en clientes de correo electrónico

Los correos electrónicos se han convertido en un recurso indispensable para el día a día de una gran multitud de personas, llegando a extraer los clientes de correos del navegador para llevarlos al fácil acceso del escritorio junto con una multitud de servicios que siguen aumentando. Una de esas nuevas funciones son los agregadores que algunos clientes de correo ya incluyen.

Los más relevantes son:

- > **Canal RSS en Outlook Express:** Outlook es uno de los más, sino el más, importante de los clientes de correo electrónico. Su función canal RSS permite la recepción de noticias con la comodidad con la que recibimos el correo.
- > **Canal RSS en Mozilla Thunderbird:** Al igual que Outlook el cliente de correo Mozilla Thunderbird ofrece un servicio de agregadores.



## 7.7.

### Directorios de canales de contenidos

Estos directorios sirven para dar a conocer los ficheros RSS publicados, son un escaparate donde los feeds se pueden registrar con el fin de obtener visualización ante los usuarios y aplicaciones usadas por estos.

Uno de estos directorios es Busca Zoom, en el cual los autores de los feeds pueden publicar sus escritos de modo que puedan ser visualizados.

Puedes acceder a esta página desde: <https://www.busca-zoom.com/>.







 [www.universae.com](http://www.universae.com)

