

- Find on Bloglines
Find on IceRocket

ttl	ttl stands for time to live. It's a number of minutes that indicates how long a channel can be cached before refreshing from the source. More info here .	<ttl>60</ttl>
image	Specifies a GIF, JPEG or PNG image that can be displayed with the channel. More info here .	
rating	The PICS rating for the channel.	
textInput	Specifies a text input box that can be displayed with the channel. More info here .	
skipHours	A hint for aggregators telling them which hours they can skip. This element contains up to 24 <hour> sub-elements whose value is a number between 0 and 23, representing a time in GMT, when aggregators, if they support the feature, may not read the channel on hours listed in the <skipHours> element. The hour beginning at midnight is hour zero.	
skipDays	A hint for aggregators telling them which days they can skip. This element contains up to seven <day> sub-elements whose value is Monday, Tuesday, Wednesday, Thursday, Friday, Saturday or Sunday. Aggregators may not read the channel during days listed in the <skipDays> element.	

<image> sub-element of <channel>

<image> is an optional sub-element of <channel>, which contains three required and three optional sub-elements.

<url> is the URL of a GIF, JPEG or PNG image that represents the channel.

<title> describes the image, it's used in the ALT attribute of the HTML tag when the channel is rendered in HTML.

<link> is the URL of the site, when the channel is rendered, the image is a link to the site. (Note, in practice the image <title> and <link> should have the same value as the channel's <title> and <link>.

Optional elements include <width> and <height>, numbers, indicating the width and height of the image in pixels. <description> contains text that is included in the TITLE attribute of the link formed around the image in the HTML rendering.

Maximum value for width is 144, default value is 88.

Maximum value for height is 400, default value is 31.

<cloud> sub-element of <channel>

<cloud> is an optional sub-element of <channel>.

It specifies a web service that supports the rssCloud interface which can be implemented in HTTP-POST, XML-RPC or SOAP 1.1.

Its purpose is to allow processes to register with a cloud to be notified of updates to the channel, implementing a lightweight publish-subscribe protocol for RSS feeds.

```
<cloud domain="rpc.sys.com" port="80" path="/RPC2" registerProcedure="myCloud.rssPleaseNotify" protocol="xml-rpc" />
```

In this example, to request notification on the channel it appears in, you would send an XML-RPC message to rpc.sys.com on port 80, with a path of /RPC2. The procedure to call is myCloud.rssPleaseNotify.

A full explanation of this element and the rssCloud interface is **here**.

<ttl> sub-element of <channel>

<ttl> is an optional sub-element of <channel>.

ttl stands for time to live. It's a number of minutes that indicates how long a channel can be cached before refreshing from the source. This makes it possible for RSS sources to be managed by a file-sharing network such as Gnutella.

Example:

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

<textInput> sub-element of <channel>

A channel may optionally contain a <textInput> sub-element, which contains four required sub-elements.

<title> -- The label of the Submit button in the text input area.

<description> -- Explains the text input area.

<name> -- The name of the text object in the text input area.

<link> -- The URL of the CGI script that processes text input requests.

The purpose of the <textInput> element is something of a mystery. You can use it to specify a search engine box. Or to allow a reader to provide feedback. Most aggregators ignore it.

Elements of <item>

A channel may contain any number of <item>s. An item may represent a "story" -- much like a story in a newspaper or magazine; if so its description is a synopsis of the story, and the link points to the full story. An item may also be complete in itself, if so, the description contains the text (entity-encoded HTML is allowed; see **examples**), and the link and title may be omitted. All elements of an item are optional, however at least one of title or description must be present.

Element	Description	Example
title	The title of the item.	Venice Film Festival Tries to Quit Sinking
link	The URL of the item.	http://nytimes.com/2004/12/07FEST.html
description	The item synopsis.	<description>Some of the most heated chatter at the Venice Film Festival this week was about the way that the arrival of the stars at the Palazzo del Cinema was being staged.</description>
author	Email address of the author of the item. More .	
category	Includes the item in one or more categories. More .	
comments	URL of a page for comments relating to the item. More .	
enclosure	Describes a media object that is attached to the item. More .	
guid	A string that uniquely identifies the item. More .	
pubDate	Indicates when the item was published. More .	
source	The RSS channel that the item came from. More .	

<source> sub-element of <item>

<source> is an optional sub-element of <item>.

Its value is the name of the RSS channel that the item came from, derived from its <title>. It has one required attribute, url, which links to the XMLization of the source.

```
<source url="http://www.tomalak.org/links2.xml">Tomalak's Realm</source>
```

The purpose of this element is to propagate credit for links, to publicize the sources of news items. It can be used in the Post command of an aggregator. It should be generated automatically when forwarding an item from an aggregator to a weblog authoring tool.

<enclosure> sub-element of <item> ↗

<enclosure> is an optional sub-element of <item>.

It has three required attributes. url says where the enclosure is located, length says how big it is in bytes, and type says what its type is, a standard MIME type.

The url must be an http url.

```
<enclosure url="http://www.scripting.com/mp3s/weatherReportSuite.mp3" length="12216320" type="audio/mpeg" />
```

A use-case narrative for this element is [here](#).

<category> sub-element of <item> ↗

<category> is an optional sub-element of <item>.

It has one optional attribute, domain, a string that identifies a categorization taxonomy.

The value of the element is a forward-slash-separated string that identifies a hierarchic location in the indicated taxonomy. Processors may establish conventions for the interpretation of categories. Two examples are provided below:

```
<category>Grateful Dead</category>
```

```
<category domain="http://www.fool.com/cusips">MSFT</category>
```

You may include as many category elements as you need to, for different domains, and to have an item cross-referenced in different parts of the same domain.

<pubDate> sub-element of <item> ↗

<pubDate> is an optional sub-element of <item>.

Its value is a **date**, indicating when the item was published. If it's a date in the future, aggregators may choose to not display the item until that date.

```
<pubDate>Sun, 19 May 2002 15:21:36 GMT</pubDate>
```

<guid> sub-element of <item> ↗

<guid> is an optional sub-element of <item>.

guid stands for globally unique identifier. It's a string that uniquely identifies the item. When present, an aggregator may choose to use this string to determine if an item is new.

```
<guid>http://some.server.com/weblogItem3207</guid>
```

There are no rules for the syntax of a guid. Aggregators must view them as a string. It's up to the source of the feed to establish the uniqueness of the string.

If the guid element has an attribute named isPermaLink with a value of true, the reader may assume that it is a permalink to the item, that is, a url that can be opened in a Web browser, that points to the full item described by the <item> element. An example:

```
<guid isPermaLink="true">http://inessential.com/2002/09/01.php#a2</guid>
```

isPermaLink is optional, its default value is true. If its value is false, the guid may not be assumed to be a url, or a url to anything in particular.

<comments> sub-element of <item> ↗

<comments> is an optional sub-element of <item>.

If present, it is the url of the comments page for the item.

```
<comments>http://ekzemplo.com/entry/4403/comments</comments>
```

More about comments [here](#).

<author> sub-element of <item> ↗

<author> is an optional sub-element of <item>.

It's the email address of the author of the item. For newspapers and magazines syndicating via RSS, the author is the person who wrote the article that the <item> describes. For collaborative weblogs, the author of the item might be different from the managing editor or webmaster. For a weblog authored by a single individual it would make sense to omit the <author> element.

```
<author>lawyer@boyer.net (Lawyer Boyer)</author>
```

Comments ↗

RSS places restrictions on the first non-whitespace characters of the data in <link> and <url> elements. The data in these elements must begin with an **IANA-registered** URI scheme, such as http://, https://, news://, mailto: and ftp://. Prior to RSS 2.0, the specification only allowed http:// and ftp://, however, in practice other URI schemes were in use by content developers and supported by aggregators. Aggregators may have limits on the URI schemes they support. Content developers should not assume that all aggregators support all schemes.

In RSS 0.91, various elements are restricted to 500 or 100 characters. There can be no more than 15 <items> in a 0.91 <channel>. There are no string-length or XML-level limits in RSS 0.92 and greater. Processors may impose their own limits, and generators may have preferences that say no more than a certain number of <item>s can appear in a channel, or that strings are limited in length.

In RSS 2.0, a provision is made for linking a channel to its identifier in a cataloging system, using the channel-level category feature, described above. For example, to link a channel to its Syndic8 identifier, include a category element as a sub-element of <channel>, with domain "Syndic8", and value the identifier for your channel in the Syndic8 database. The appropriate category element for Scripting News would be <category domain="Syndic8">1765</category>.

A frequently asked question about <guid>s is how do they compare to <link>s. Aren't they the same thing? Yes, in some content systems, and no in others. In some systems, <link> is a permalink to a weblog item. However, in other systems, each <item> is a synopsis of a longer article, <link> points to the article, and <guid> is the permalink to the weblog entry. In all cases, it's recommended that you provide the guid, and if possible make it a permalink. This enables aggregators to not repeat items, even if there have been editing changes.

If you have questions about the RSS 2.0 format, please post them on the **RSS-Public** mailing list. The list, maintained by the RSS Advisory Board, serves as a support resource for users, authors and developers who are creating and using content in the format.

Extending RSS ↗

RSS originated in 1999, and has strived to be a simple, easy to understand format, with relatively modest goals. After it became a popular format, developers wanted to extend it using modules defined in namespaces, as **specified** by the W3C.

RSS 2.0 adds that capability, following a simple rule. A RSS feed may contain elements and attributes not described on this page, only if those elements and attributes are defined in a namespace.

The elements defined in this document are not themselves members of a namespace, so that RSS 2.0 can remain compatible with previous versions in the following sense -- a version 0.91 or 0.92 file is also a valid 2.0 file. If the elements of RSS 2.0 were in a namespace, this constraint would break, a version 0.9x file *would not be* a valid 2.0 file.

Roadmap ↗

RSS is by no means a perfect format, but it is very popular and widely supported. Having a settled spec is something RSS has needed for a long time. The purpose of this work is to help it become a unchanging thing, to foster growth in the market that is developing around it, and to clear the path for innovation in new syndication formats. Therefore, the RSS spec is, for all practical purposes, frozen at version 2.0.1. We anticipate possible 2.0.2 or 2.0.3 versions, etc. only for the purpose of clarifying the specification, not for adding new features to the format. Subsequent work should happen in modules, using namespaces, and in completely new syndication formats, with new names.

License and authorship ↗

