sitemaps.org

Sitemaps XML format

This document describes the XML schema for the Sitemap protocol.

The Sitemap protocol format consists of XML tags. All data values in a Sitemap must be entity-escaped. The file itself must be UTF-8 encoded.

The Sitemap must:

- Begin with an opening <urlset> tag and end with a closing </urlset> tag.
- Specify the namespace (protocol standard) within the <urlset> tag.
- Include a <url> entry for each URL, as a parent XML tag.
- Include a <loc> child entry for each <url> parent tag.

All other tags are optional. Support for these optional tags may vary among search engines. Refer to each search engine's documentation for details.

Also, all URLs in a Sitemap must be from a single host, such as www.example.com or store.example.com. For further details, refer the Sitemap file location

Jump to:
XML tag definitions
Entity escaping
Using Sitemap index files
Other Sitemap formats
Sitemap file location
Validating your Sitemap
Extending the Sitemaps protocol
Informing search engine crawlers

Sample XML Sitemap

The following example shows a Sitemap that contains just one URL and uses all optional tags. The optional tags are in italics.

```
<?xml version="1.0" encoding="UTF-8"?>
<urlset xmlns="http://www.sitemaps.org/schemas/sitemap/0.9">
  <url>
    <loc>http://www.example.com/</loc>
    <lastmod>2005-01-01</lastmod>
    <changefreq>monthly</changefreq>
    <priority>0.8</priority>
  </url>
</urlset>
```

Also see our example with multiple URLs.

XML tag definitions

The available XML tags are described below.

Attribute		Description
<urlset>	required	Encapsulates the file and references the current protocol standard.

<url>	required	==Parent tag for each URL entry.== The remaining tags are children of this tag.
<loc>	required	==URL of the page. This URL must begin with the protocol (such as http) and end with a trailing slash, if your web server requires it.== This value must be less than 2,048 characters.
<lastmod>	optional	The date of last modification of the file. This date should be in W3C Datetime format. This format allows you to omit the time portion, if desired, and use YYYY-MM-DD. Note that this tag is separate from the If-Modified-Since (304) header the server can return, and search engines may use the information from both sources differently.
<changefreq>	optional	How frequently the page is likely to change. This value provides general information to search engines and may not correlate exactly to how often they crawl the page. Valid values are: - always - hourly - daily - weekly - monthly - yearly - never The value "always" should be used to describe documents that change each time they are accessed. The value "never" should be used to describe archived URLs. Please note that the value of this tag is considered a *hint* and not a command. Even though search engine crawlers may consider this information when making decisions, they may crawl pages marked "hourly" less frequently than that, and they may crawl pages marked "yearly" more frequently than that. Crawlers may periodically crawl pages marked "never" so that they can handle unexpected changes to those pages.

`<priority>`	optional	The priority of this URL relative to other URLs on your site. Valid values range from 0.0 to 1.0. This value does not affect how your pages are compared to pages on other sites—it only lets the search engines know which pages you deem most important for the crawlers. The default priority of a page is 0.5. Please note that the priority you assign to a page is not likely to influence the position of your URLs in a search engine's result pages. Search engines may use this information when selecting between URLs on the same site, so you can use this tag to increase the likelihood that your most important pages are present in a search index. Also, please note that assigning a high priority to all of the URLs on your site is not likely to help you. Since the priority is relative, it is only used to select between URLs on your site.

Back to top

Entity escaping

Your Sitemap file must be UTF-8 encoded (you can generally do this when you save the file). As with all XML files, any data values (including URLs) must use entity escape codes for the characters listed in the table below.

Character		Escape Code
Ampersand	&	&
Single Quote	'	'
Double Quote	"	"
Greater Than	>	>
Less Than	<	<

In addition, all URLs (including the URL of your Sitemap) must be URL-escaped and encoded for readability by the web server on which they are located. However, if you are using any sort of script, tool, or log file to generate your URLs (anything except typing them in by hand), this is usually already done for you. Please check to make sure that your URLs follow the RFC-3986 standard for URIs, the RFC-3987 standard for IRIs, and the XML standard.

Below is an example of a URL that uses a non-ASCII character (ü), as well as a character that requires entity escaping (&):

    http://www.example.com/ümlat.php&q=name

Below is that same URL, ISO-8859-1 encoded (for hosting on a server that uses that encoding) and URL escaped:

    http://www.example.com/%FCmlat.php&q=name

Below is that same URL, UTF-8 encoded (for hosting on a server that uses that encoding) and URL escaped:

```
http://www.example.com/%C3%BCmlat.php&q=name
```

Below is that same URL, but also entity escaped:

```
http://www.example.com/%C3%BCmlat.php&amp;q=name
```

Sample XML Sitemap

The following example shows a Sitemap in XML format. The Sitemap in the example contains a small number of URLs, each using a different set of optional parameters.

```xml
<?xml version="1.0" encoding="UTF-8"?>
<urlset xmlns="http://www.sitemaps.org/schemas/sitemap/0.9">
  <url>
     <loc>http://www.example.com/</loc>
     <lastmod>2005-01-01</lastmod>
     <changefreq>monthly</changefreq>
     <priority>0.8</priority>
  </url>
  <url>
     <loc>http://www.example.com/catalog?item=12&amp;desc=vacation_hawaii</loc>
     <changefreq>weekly</changefreq>
  </url>
  <url>
     <loc>http://www.example.com/catalog?item=73&amp;desc=vacation_new_zealand</loc>
     <lastmod>2004-12-23</lastmod>
     <changefreq>weekly</changefreq>
  </url>
  <url>
     <loc>http://www.example.com/catalog?item=74&amp;desc=vacation_newfoundland</loc>
     <lastmod>2004-12-23T18:00:15+00:00</lastmod>
     <priority>0.3</priority>
  </url>
  <url>
     <loc>http://www.example.com/catalog?item=83&amp;desc=vacation_usa</loc>
     <lastmod>2004-11-23</lastmod>
  </url>
</urlset>
```

Back to top

Using Sitemap index files (to group multiple sitemap files)

You can provide multiple Sitemap files, but each Sitemap file that you provide must have no more than 50,000 URLs and must be no larger than 10MB (10,485,760 bytes). If you would like, you may compress your Sitemap files using gzip to reduce your bandwidth requirement; however the sitemap file once uncompressed must be no larger than 10MB. If you want to list more than 50,000 URLs, you must create multiple Sitemap files.

If you do provide multiple Sitemaps, you should then list each Sitemap file in a Sitemap index file. Sitemap index files may not list more than 50,000 Sitemaps and must be no larger than 10MB (10,485,760 bytes) and can be compressed. You can have more than one Sitemap index file. The XML format of a Sitemap index file is very similar to the XML format of a Sitemap file.

The Sitemap index file must:

- Begin with an opening <sitemapindex> tag and end with a closing </sitemapindex> tag.
- Include a <sitemap> entry for each Sitemap as a parent XML tag.
- Include a <loc> child entry for each <sitemap> parent tag.

The optional <lastmod> tag is also available for Sitemap index files.

Note: A Sitemap index file can only specify Sitemaps that are found on the same site as the Sitemap index file. For example, http://www.yoursite.com/sitemap_index.xml can include Sitemaps on http://www.yoursite.com but not on http://www.example.com or http://yourhost.yoursite.com. As with Sitemaps, your Sitemap index file must be UTF-8 encoded.

Sample XML Sitemap Index

The following example shows a Sitemap index that lists two Sitemaps:

```
<?xml version="1.0" encoding="UTF-8"?>
<sitemapindex xmlns="http://www.sitemaps.org/schemas/sitemap/0.9">
   <sitemap>
      <loc>http://www.example.com/sitemap1.xml.gz</loc>
      <lastmod>2004-10-01T18:23:17+00:00</lastmod>
   </sitemap>
   <sitemap>
      <loc>http://www.example.com/sitemap2.xml.gz</loc>
      <lastmod>2005-01-01</lastmod>
   </sitemap>
</sitemapindex>
```

Note: Sitemap URLs, like all values in your XML files, must be entity escaped.

Sitemap Index XML Tag Definitions

Attribute		Description
<sitemapindex>	required	Encapsulates information about all of the Sitemaps in the file.
<sitemap>	required	Encapsulates information about an individual Sitemap.
<loc>	required	Identifies the location of the Sitemap. This location can be a Sitemap, an Atom file, RSS file or a simple text file.
<lastmod>	optional	Identifies the time that the corresponding Sitemap file was modified. It does not correspond to the time that any of the pages listed in that Sitemap were changed. The value for the lastmod tag should be in W3C Datetime format. By providing the last modification timestamp, you enable search engine crawlers to retrieve only a subset of the Sitemaps in the index i.e. a crawler may only retrieve Sitemaps that were modified

		since a certain date. This incremental Sitemap fetching mechanism allows for the rapid discovery of new URLs on very large sites.

Other Sitemap formats

The Sitemap protocol enables you to provide details about your pages to search engines, and we encourage its use since you can provide additional information about site pages beyond just the URLs. However, in addition to the XML protocol, we support RSS feeds and text files, which provide more limited information.

Syndication feed

You can provide an RSS (Real Simple Syndication) 2.0 or Atom 0.3 or 1.0 feed. Generally, you would use this format only if your site already has a syndication feed. Note that this method may not let search engines know about all the URLs in your site, since the feed may only provide information on recent URLs, although search engines can still use that information to find out about other pages on your site during their normal crawling processes by following links inside pages in the feed. Make sure that the feed is located in the highest-level directory you want search engines to crawl. Search engines extract the information from the feed as follows:

- <link> field - indicates the URL
- modified date field (the <pubDate> field for RSS feeds and the <modified> date for Atom feeds) - indicates when each URL was last modified. Use of the modified date field is optional.

Text file

You can provide a simple text file that contains one URL per line. The text file must follow these guidelines:

- The text file must have one URL per line. The URLs cannot contain embedded new lines.
- You must fully specify URLs, including the http.
- Each text file can contain a maximum of 50,000 URLs and must be no larger than 10MB (10,485,760 bytes). If you site includes more than 50,000 URLs, you can separate the list into multiple text files and add each one separately.
- The text file must use UTF-8 encoding. You can specify this when you save the file (for instance, in Notepad, this is listed in the Encoding menu of the Save As dialog box).
- The text file should contain no information other than the list of URLs.
- The text file should contain no header or footer information.
- If you would like, you may compress your Sitemap text file using gzip to reduce your bandwidth requirement.
- You can name the text file anything you wish. Please check to make sure that your URLs follow the RFC-3986 standard for URIs, the RFC-3987 standard for IRIs
- You should upload the text file to the highest-level directory you want search engines to crawl and make sure that you don't list URLs in the text file that are located in a higher-level directory.

Sample text file entries are shown below.

```
http://www.example.com/catalog?item=1
http://www.example.com/catalog?item=11
```

Sitemap file location

The location of a Sitemap file determines the set of URLs that can be included in that Sitemap. A Sitemap file located at http://example.com/catalog/sitemap.xml can include any URLs starting with http://example.com/catalog/ but can not include URLs starting with http://example.com/images/.

If you have the permission to change http://example.org/path/sitemap.xml, it is assumed that you also have permission to provide

information for URLs with the prefix http://example.org/path/. Examples of URLs considered valid in http://example.com/catalog/
sitemap.xml include:

```
http://example.com/catalog/show?item=23
http://example.com/catalog/show?item=233&user=3453
```

URLs not considered valid in http://example.com/catalog/sitemap.xml include:

```
http://example.com/image/show?item=23
http://example.com/image/show?item=233&user=3453
https://example.com/catalog/page1.php
```

Note that this means that all URLs listed in the Sitemap must use the same protocol (http, in this example) and reside on the same
host as the Sitemap. For instance, if the Sitemap is located at http://www.example.com/sitemap.xml, it can't include URLs from
http://subdomain.example.com.

URLs that are not considered valid are dropped from further consideration. It is strongly recommended that you place your Sitemap
at the root directory of your web server. For example, if your web server is at example.com, then your Sitemap index file would be at
http://example.com/sitemap.xml. In certain cases, you may need to produce different Sitemaps for different paths (e.g., if security
permissions in your organization compartmentalize write access to different directories).

If you submit a Sitemap using a path with a port number, you must include that port number as part of the path in each URL listed in
the Sitemap file. For instance, if your Sitemap is located at http://www.example.com:100/sitemap.xml, then each URL listed in the
Sitemap must begin with http://www.example.com:100.

Sitemaps & Cross Submits

To submit Sitemaps for multiple hosts from a single host, you need to "prove" ownership of the host(s) for which URLs are being
submitted in a Sitemap. Here's an example. Let's say that you want to submit Sitemaps for 3 hosts:

```
www.host1.com with Sitemap file sitemap-host1.xml
www.host2.com with Sitemap file sitemap-host2.xml
www.host3.com with Sitemap file sitemap-host3.xml
```

Moreover, you want to place all three Sitemaps on a single host: www.sitemaphost.com. So the Sitemap URLs will be:

```
http://www.sitemaphost.com/sitemap-host1.xml
http://www.sitemaphost.com/sitemap-host2.xml
http://www.sitemaphost.com/sitemap-host3.xml
```

By default, this will result in a "cross submission" error since you are trying to submit URLs for www.host1.com through a Sitemap
that is hosted on www.sitemaphost.com (and same for the other two hosts). One way to avoid the error is to prove that you own (i.e.
have the authority to modify files) www.host1.com. You can do this by modifying the robots.txt file on www.host1.com to point to
the Sitemap on www.sitemaphost.com.

In this example, the robots.txt file at http://www.host1.com/robots.txt would contain the line "Sitemap: http://www.sitemaphost.
com/sitemap-host1.xml". By modifying the robots.txt file on www.host1.com and having it point to the Sitemap on www.sitemaphost.
com, you have implicitly proven that you own www.host1.com. In other words, whoever controls the robots.txt file on www.host1.
com trusts the Sitemap at http://www.sitemaphost.com/sitemap-host1.xml to contain URLs for www.host1.com. The same process
can be repeated for the other two hosts.

Now you can submit the Sitemaps on www.sitemaphost.com.

When a particular host's robots.txt, say http://www.host1.com/robots.txt, points to a Sitemap or a Sitemap index on another host; it
is expected that for each of the target Sitemaps, such as http://www.sitemaphost.com/sitemap-host1.xml, all the URLs belong to
the host pointing to it. This is because, as noted earlier, a Sitemap is expected to have URLs from a single host only.

Back to top

Validating your Sitemap

The following XML schemas define the elements and attributes that can appear in your Sitemap file. You can download this schema from the links below:

For Sitemaps: http://www.sitemaps.org/schemas/sitemap/0.9/sitemap.xsd

For Sitemap index files: http://www.sitemaps.org/schemas/sitemap/0.9/siteindex.xsd

There are a number of tools available to help you validate the structure of your Sitemap based on this schema. You can find a list of XML-related tools at each of the following locations:

http://www.w3.org/XML/Schema#Tools

http://www.xml.com/pub/a/2000/12/13/schematools.html

In order to validate your Sitemap or Sitemap index file against a schema, the XML file will need additional headers as shown below.

Sitemap:

```
<?xml version='1.0' encoding='UTF-8'?>
<urlset xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
      xsi:schemaLocation="http://www.sitemaps.org/schemas/sitemap/0.9 http://www.sitemaps.org/schemas/sitemap/0.9/sitemap.xsd"
      xmlns="http://www.sitemaps.org/schemas/sitemap/0.9">
  <url>
    ...
  </url>
</urlset>
```

Sitemap index file:

```
<?xml version='1.0' encoding='UTF-8'?>
<sitemapindex xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
      xsi:schemaLocation="http://www.sitemaps.org/schemas/sitemap/0.9 http://www.sitemaps.org/schemas/sitemap/0.9/siteindex.xsd"
      xmlns="http://www.sitemaps.org/schemas/sitemap/0.9">
  <sitemap>
    ...
  </sitemap>
</sitemapindex>
```

Back to top

Extending the Sitemaps protocol

You can extend the Sitemaps protocol using your own namespace. Simply specify this namespace in the root element. For example:

```
<?xml version='1.0' encoding='UTF-8'?>
<urlset xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
      xsi:schemaLocation="http://www.sitemaps.org/schemas/sitemap/0.9 http://www.sitemaps.org/schemas/sitemap/0.9/sitemap.xsd"
      xmlns="http://www.sitemaps.org/schemas/sitemap/0.9"
      xmlns:example="http://www.example.com/schemas/example_schema"> <!-- namespace extension -->
  <url>
    <example:example_tag>
      ...
    </example:example_tag>
    ...
  </url>
</urlset>
```

Back to top

Informing search engine crawlers

Once you have created the Sitemap file and placed it on your webserver, you need to inform the search engines that support this protocol of its location. You can do this by:

- submitting it to them via the search engine's submission interface
- specifying the location in your site's robots.txt file
- sending an HTTP request

The search engines can then retrieve your Sitemap and make the URLs available to their crawlers.

Submitting your Sitemap via the search engine's submission interface

To submit your Sitemap directly to a search engine, which will enable you to receive status information and any processing errors, refer to each search engine's documentation.

Specifying the Sitemap location in your robots.txt file

You can specify the location of the Sitemap using a robots.txt file. To do this, simply add the following line including the full URL to the sitemap:

```
Sitemap: http://www.example.com/sitemap.xml
```

This directive is independent of the user-agent line, so it doesn't matter where you place it in your file. If you have a Sitemap index file, you can include the location of just that file. You don't need to list each individual Sitemap listed in the index file.

You can specify more than one Sitemap file per robots.txt file.

```
Sitemap: http://www.example.com/sitemap-host1.xml
Sitemap: http://www.example.com/sitemap-host2.xml
```

Submitting your Sitemap via an HTTP request

To submit your Sitemap using an HTTP request (replace <searchengine_URL> with the URL provided by the search engine), issue your request to the following URL:

```
<searchengine_URL>/ping?sitemap=sitemap_url
```

For example, if your Sitemap is located at http://www.example.com/sitemap.gz, your URL will become:

```
<searchengine_URL>/ping?sitemap=http://www.example.com/sitemap.gz
```

URL encode everything after the /ping?sitemap=:

```
<searchengine_URL>/ping?sitemap=http%3A%2F%2Fwww.yoursite.com%2Fsitemap.gz
```

You can issue the HTTP request using wget, curl, or another mechanism of your choosing. A successful request will return an HTTP 200 response code; if you receive a different response, you should resubmit your request. The HTTP 200 response code only indicates that the search engine has received your Sitemap, not that the Sitemap itself or the URLs contained in it were valid. An easy way to do this is to set up an automated job to generate and submit Sitemaps on a regular basis.
Note: If you are providing a Sitemap index file, you only need to issue one HTTP request that includes the location of the Sitemap index file; you do not need to issue individual requests for each Sitemap listed in the index.

Back to top

Excluding content

The Sitemaps protocol enables you to let search engines know what content you would like indexed. To tell search engines the content you don't want indexed, use a robots.txt file or robots meta tag. See robotstxt.org for more information on how to exclude content from search engines.

Back to top

Last Updated: 27 February 2008

Terms and conditions