

Sitemap
Plug-in Guide
2.0.0-3

Table of Contents

1 Introduction	3
1.1 How It Works	3
1.1.1 Sitemap Extensions	4
1.1.2 How Sitemaps are Generated	6
2 Installation	8
2.1 Conventions	8
2.2 Installing the Plug-in	8
2.3 Verify the Installation	9
3 Configuration	10
3.1 Editing the content-type Resource	10
3.2 Editing the feature resource	12
3.2.1 Google News-related properties	12
3.2.2 Google Video-related properties	13
3.3 Editing SitemapConfig.properties	14
3.4 Exposing The Sitemaps	14
3.5 Configuring Section Parameters	15
3.6 Configuring Solr	15
4 Administration	16
4.1 Sitemap Caching	16

1 Introduction

The Content Engine's Sitemap plug-in generates sitemaps for Escenic sites. Sitemaps are XML files that contain links to all the resources on a web site that you want to be indexed by search engines. They provide an efficient means of directing search engines to all the important content on your site.

1.1 How It Works

The Sitemap plug-in generates sitemaps in the **sitemaps.org** XML format specified in <http://www.sitemaps.org/protocol.html>. This is what a (very small) **sitemap.org** sitemap document looks like:

```
<?xml version="1.0" encoding="UTF-8"?>
<urlset xmlns="http://www.sitemaps.org/schemas/sitemap/0.9">
  <url>
    <loc>http://www.example.com/incoming/article51.ece</loc>
    <lastmod>2013-05-31T11:51:13+06:00</lastmod>
  </url>
  <url>
    <loc>http://www.example.com/incoming/article42.ece</loc>
    <lastmod>2013-05-31T11:51:23+06:00</lastmod>
  </url>
</urlset>
```

The plug-in can generate two basic types of sitemap:

Aggregated sitemap

An aggregated sitemap contains the URLs of all selected content items that are in a published state at the time the sitemap is generated. This kind of sitemap is only really intended to be generated one time, when a site is first published and you want ensure that the entire site is indexed. The idea is that you explicitly request generation of the sitemap yourself and then upload it to the search engines you are interested in.

Update sitemap

An update sitemap only contains the URLs of recently published content items that have been published recently (by default over the last 72 hours). The idea is that you publish the URL of this sitemap in your site's **robots.txt** file so that it can be found by search engine indexers, which periodically visit it and index all the listed URLs. Alternatively you can control the process yourself by creating an application or **cron** job that actively posts it to the search engines you are interested in at intervals.

Both types of sitemap have exactly the same structure, the only difference is the number of entries they contain.

In order to prevent sitemap documents becoming unmanageably large, the **sitemaps.org** standard allows sitemaps to be split into multiple documents that are then referenced by a master sitemap index. The Sitemap plug-in makes use of this feature. It generates one sitemap index per Escenic publication, which in turn references one sitemap document for every content type that you choose to include. Here is a small example of a sitemap index:

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

```

<sitemapindex xmlns="http://www.sitemaps.org/schemas/sitemap/0.9">
  <sitemap>
    <loc>http://www.example.com/sitemap/sections.xml</loc>
    <lastmod>2013-05-31T12:07:35+06:00</lastmod>
  </sitemap>
  <sitemap>
    <loc>http://www.example.com/sitemap/news.xml</loc>
    <lastmod>2013-05-31T12:07:35+06:00</lastmod>
  </sitemap>
  <sitemap>
    <loc>http://www.example.com/sitemap/review.xml</loc>
    <lastmod>2013-05-31T12:07:35+06:00</lastmod>
  </sitemap>
  <sitemap>
    <loc>http://www.example.com/sitemap/video.xml</loc>
    <lastmod>2013-05-31T12:07:35+06:00</lastmod>
  </sitemap>
</sitemapindex>

```

If number of articles of a content type exceeds entry per sitemap value which is defined in **SitemapConfig.properties** then sitemap documents are generated based on the ratio of entry per sitemap value for every content type. Here is a small example of a sitemap index which contains multiple sitemap documents for a single content type:

```

<?xml version="1.0" encoding="UTF-8"?>
  <sitemapindex xmlns="http://www.sitemaps.org/schemas/sitemap/0.9">
    <sitemap>
      <loc>http://www.example.com/sitemap/sections.xml</loc>
      <lastmod>2013-05-31T12:07:35+06:00</lastmod>
    </sitemap>
    <sitemap>
      <loc>http://www.example.com/sitemap/news/1.xml</loc>
      <lastmod>2013-05-31T12:07:35+06:00</lastmod>
    </sitemap>
    <sitemap>
      <loc>http://www.example.com/sitemap/news/2.xml</loc>
      <lastmod>2013-05-31T12:07:35+06:00</lastmod>
    </sitemap>
    <sitemap>
      <loc>http://www.example.com/sitemap/review.xml</loc>
      <lastmod>2013-05-31T12:07:35+06:00</lastmod>
    </sitemap>
    <sitemap>
      <loc>http://www.example.com/sitemap/video.xml</loc>
      <lastmod>2013-05-31T12:07:35+06:00</lastmod>
    </sitemap>
  </sitemapindex>

```

You choose which content types you want to be included by adding **seo:enabled** elements to content types in your publication **content-type** resources (see [section 3.1](#)).

1.1.1 Sitemap Extensions

The sitemaps generated by the Sitemap plug-in can, if required, contain extension elements. The extension elements allow you specify additional information about certain types of content that will be used by some specialized search engines. The extension elements belong to different namespaces from

the standard [sitemap.org](http://www.sitemaps.org) elements and will therefore be ignored by any search engines that do not recognise them.

The sitemap extensions that are currently supported are defined and used by Google:

Google News sitemaps

This extension lets you identify content that you want to be submitted to Google News. See [section 1.1.1.1](#) for an example of a sitemap containing these extensions.

Google Video sitemaps

This extension lets you identify content that you want to be submitted to Google Video. See [section 1.1.1.2](#) for an example of a sitemap containing these extensions.

You can activate these extensions for some content types and not for others (see [section 3.1](#) for details). You would, for example, only activate the Google Video extensions for content types that contain video clips.

1.1.1.1 Google News Sitemaps

Here is an example sitemap document that contains Google News extension elements:

```
<?xml version="1.0" encoding="UTF-8"?>
<urlset xmlns="http://www.sitemaps.org/schemas/sitemap/0.9" xmlns:image="http://
www.google.com/schemas/sitemap-image/1.1" xmlns:news="http://www.google.com/schemas/
sitemap-news/0.9">
  <url>
    <loc>http://www.example.com/incoming/article51.ece</loc>
    <lastmod>2013-05-31T11:51:13+06:00</lastmod>
    <news:news>
      <news:publication>
        <news:name>News Site</news:name>
        <news:language>en</news:language>
      </news:publication>
      <news:publication_date>2013-05-10T11:23:22+06:00</news:publication_date>
      <news:title>Article May 10</news:title>
    </news:news>
    <image:image>
      <image:loc>http://www.example.com/incoming/article50.ece/alternates/
WideBig/2.jpg</image:loc>
      <image:title>2.jpg</image:title>
    </image:image>
  </url>
  <url>
    <loc>http://www.example.com/incoming/article42.ece</loc>
    <lastmod>2013-05-31T11:51:23+06:00</lastmod>
    <news:news>
      <news:publication>
        <news:name>News Site</news:name>
        <news:language>en</news:language>
      </news:publication>
      <news:publication_date>2013-05-07T13:09:28+06:00</news:publication_date>
      <news:title>Article May 7</news:title>
    </news:news>
  </url>
</urlset>
```

For detailed information about this sitemap extension, see <http://support.google.com/webmasters/bin/answer.py?hl=en&answer=183668&topic=8476&ctx=topic#2>.

1.1.1.2 Google Video Sitemaps

Here is an example sitemap document that contains Google Video extension elements:

```
<?xml version="1.0" encoding="UTF-8"?>
<urlset xmlns="http://www.sitemaps.org/schemas/sitemap/0.9" xmlns:video="http://
www.google.com/schemas/sitemap-video/1.1">
  <url>
    <loc>http://www.example.com/incoming/article97.ece</loc>
    <lastmod>2013-05-30T10:19:15+06:00</lastmod>
    <video:video>
      <video:thumbnail_loc>http://www.example.com/incoming/article102.ece/
alternates/SmallSquare/Test%2029th%20May</video:thumbnail_loc>
      <video:title>Test 29th May</video:title>
      <video:description>Test 29th May</video:description>
      <video:content_loc>http://www.example.com/video10430.mp4</video:content_loc>
      <video:publication_date>2013-05-30T10:19:15+06:00</video:publication_date>
      <video:live>No</video:live>
    </video:video>
  </url>
  <url>
    <loc>http://www.example.com/incoming/article72.ece</loc>
    <lastmod>2013-05-31T11:34:28+06:00</lastmod>
    <video:video>
      <video:thumbnail_loc>http://www.example.com/incoming/article78.ece/
alternates/SmallSquare/Test%2024th%20May</video:thumbnail_loc>
      <video:title>Test 24th May</video:title>
      <video:description>Summary</video:description>
      <video:content_loc>http://www.example.com/video10137.mp4</video:content_loc>
      <video:publication_date>2013-05-24T10:47:30+06:00</video:publication_date>
      <video:live>No</video:live>
    </video:video>
  </url>
</urlset>
```

For detailed information about this sitemap extension, see <http://support.google.com/webmasters/bin/answer.py?hl=en&answer=183668&topic=8476&ctx=topic#2>.

1.1.2 How Sitemaps are Generated

The Sitemap plug-in does not **actively** generate sitemaps: it is simply a webapp that responds to certain HTTP requests. It will generate a sitemap in response to an **HTTP GET** request directed to one of four specific URLs, for example:

<http://your-publication/sitemap/archive.xml>

The plug-in generates and returns an aggregated sitemap containing the URLs of all the content items belonging to the content types for which you have enabled sitemap generation.

<http://your-publication/sitemap/update.xml>

The plug-in generates and returns a sitemap containing the URLs of all content items belonging to the content types for which you have enabled sitemap generation, and that were published in the last 72 hours. If you have enabled Google News or Google Video extensions for any content types, however, then they are omitted from this sitemap, and instead included in one of the two sitemaps listed below.

`http://your-publication/sitemap/googlenews.xml`

The plug-in generates and returns a sitemap containing the URLs of all the content items belonging to content types for which you have enabled the generation of Google News extended sitemaps, and that were published in the last 72 hours.

`http://your-publication/sitemap/googlevideo.xml`

The plug-in generates and returns a sitemap containing the URLs of all the content items belonging to content types for which you have enabled the generation of Google Video extended sitemaps, and that were published in the last 72 hours.

Once you have installed and configured the plug-in, nothing happens until a request for one of the above index documents is received. Typically you would:

1. Submit a request yourself for **`http://your-publication/sitemap/archive.xml`**.
2. Submit requests for all the sitemap URLs listed in the returned index document.
3. Post the returned sitemap documents to the search engines you want to index your site.
4. Add the URL **`http://your-publication/sitemap/update.xml`** to your **`robots.txt`** file.
5. If you have enabled Google News sitemap extensions for any of your content types, add the URL **`http://your-publication/sitemap/googlenews.xml`** to your **`robots.txt`** file.
6. If you have enabled Google Video sitemap extensions for any of your content types, add the URL **`http://your-publication/sitemap/googlevideo.xml`** to your **`robots.txt`** file.

Then each time a search engine indexer reads your **`robots.txt`** file it will:

1. Send requests for all the sitemap indexes listed there.
2. Send requests for all the sitemap URLs listed in the returned index documents.
3. Send requests for all the recently published content items listed in the returned sitemaps and index them.

If you have enabled Google News or Google Video sitemap extensions for any of your content types, then you also need to submit the URLs of your extended sitemap indexes (**`http://your-publication/sitemap/googlenews.xml`** and **`http://your-publication/sitemap/googlevideo.xml`**) to Google's Webmaster Tools. This is necessary since Google does not use the standard **`robots.txt`** mechanism for indexing news and videos. For more information about this, see <http://support.google.com/webmasters/bin/answer.py?hl=en&answer=183668&topic=8476&ctx=topic#2>.

2 Installation

The following preconditions must be met before you can install the Sitemap 2.0.0-3 plug-in:

- The Content Engine and Escenic assembly tool have been installed as described in the **Escenic Content Engine Installation Guide** and are in working order.
- You have the required plug-in distribution file `seo-2.0.0-3.zip`.

The Sitemap 2.0.0-3 plug in depends in general on Content Engine 5.4, and will work with version 5.4 or any later version of the Content Engine.

2.1 Conventions

The instructions in the following section assume that you have a standard Content Engine installation, as described in the **Escenic Content Engine Installation Guide**. *escenic-home* is used to refer to the `/opt/escenic` folder under which both the Content Engine itself and all plug-ins are installed.

The Content Engine and the software it depends on may be installed on one or several host machines depending on the type of installation required. In order to unambiguously identify the machines on which various installation actions must be carried out, the **Escenic Content Engine Installation Guide** defines a set of special host names that are used throughout the manual.

Some of these names are also used here:

assembly-host

The machine used to assemble the various Content Engine components into an enterprise archive or .EAR file.

engine-host

The machine(s) used to host application servers and Content Engine instances.

editorial-host

engine-host(s) that are used solely for (internal) editorial purposes.

The host names always appear in a bold typeface. If you are installing everything on one host you can, of course, ignore them: you can just do everything on the same machine. If you are creating a larger multi-host installation, then they should help ensure that you do things in the right places.

2.2 Installing the Plug-in

Installing the Sitemap plug-in involves the following steps:

1. Log in as **escenic** on your **assembly-host**.
2. Download the Sitemap distribution from the Escenic Technet web site (<http://technet.escenic.com>). If you have a multi-host installation with shared folders as described in the **Escenic Content Engine Installation Guide**, then it is a good idea to download the distribution to your shared `/mnt/download` folder:

```
| $ cd /mnt/download
```



```
$ wget https://user:password@maven.escenic.com/com/escenic/plugins/seo/seo/2.0.0-3/seo-2.0.0-3.zip
```

Otherwise, download it to some temporary location of your choice.

3. If the folder `/opt/escenic/engine/plugins` does not already exist, create it:

```
$ mkdir /opt/escenic/engine/plugins
```

4. Unpack the Sitemap distribution file:


```
$ cd /opt/escenic/engine/plugins
$ unzip /mnt/download/seo-2.0.0-3.zip
```

This will result in the creation of an `/opt/escenic/engine/plugins/seo` folder.

5. Log in as **escenic** on your **assembly-host**.
6. Run the **ece** script to re-assemble your Content Engine applications.

```
$ ece assemble
```

This generates an EAR file (`/var/cache/escenic/engine.ear`) that you can deploy on all your **engine-hosts**.

7.  If you have a single-host installation, then skip this step.

On each **engine-host** you wish to run the Sitemap plug-in, copy `/var/cache/escenic/engine.ear` from the **assembly-host**. If you have installed an SSH server on the **assembly-host** and SSH clients on your **engine-hosts**, then you can do this as follows:

```
$ scp -r escenic@assembly-host-ip-address:/var/cache/escenic/engine.ear /var/cache/escenic/
```

where `assembly-host-ip-address` is the host name or IP address of your **assembly-host**.

8. On each **engine-host** you wish to run the Sitemap plug-in, deploy the EAR file and restart the Content Engine by entering:

```
$ ece deploy
$ ece restart
```

2.3 Verify the Installation

To verify the status of the Sitemap plug-in, open the Escenic Admin web application (usually located at `http://server/escenic-admin`) and click on **View installed plug-ins**. The status of all currently installed plug-ins is shown here, and indicated as follows:



The plug-in is correctly installed.



The plug-in is not correctly installed.

3 Configuration

3.1 Editing the content-type Resource

To create sitemaps for an Escenic publication, all you need to do is add `seo:enabled` elements to the content type definitions in the publication's `content-type` resource. You must add an `seo:enabled` element to each content type for which you want a sitemap document to be generated.

For general information about the `content-type` resource and content type editing, see the **Escenic Content Engine Resource Reference**.

An `seo:enabled` element must be added as a child of a `content-type` element, and must belong to the namespace `http://xmlns.escenic.com/2013/seo`:

```
<content-types ...
  xmlns:seo="http://xmlns.escenic.com/2013/seo"
  ...>
  ...
  <content-type name="news">
    ...
    <seo:enabled/>
    ...
  </content-type>
  ...
</content-types>
```

If the `seo:enabled` element is specified without a `type` attribute as above, then the generated sitemap document will contain no extension elements.

If you include a `type="news"` attribute in a content type's `seo:enabled` element:

```
<content-types ...
  xmlns:seo="http://xmlns.escenic.com/2013/seo"
  ...>
  ...
  <content-type name="news">
    ...
    <seo:enabled type="news"/>
    ...
  </content-type>
  ...
</content-types>
```

then Google News extension elements will be included in the generated sitemap document.

If you include a `type="video"` attribute in a content type's `seo:enabled` element:

```
<content-types ...
  xmlns:seo="http://xmlns.escenic.com/2013/seo"
  ...>
  ...
  <content-type name="video">
```

```

...
<seo:enabled type="video" video-description="body"/>
...
<field mime-type="application/xhtml+xml" type="basic" name="body">
  <ui:label>Description</ui:label>
</field>
...
</content-type>
...
</content-types>

```

then Google Video extension elements will be included in the generated sitemap document. Note that in this case the element also has a **video-description** element that references one of the content-item's **field** elements. You should reference the field that contains the best description of the video content.

It is not possible to generate a sitemap that contains both Google News and Google video extension elements.

Include images in Google News/Video sitemaps

Both the Google News and Google Video extensions allow you to include information about images. The method is the same in both cases as shown in the following example:

```

<content-types ...
  <xmlns:seo="http://xmlns.escenic.com/2013/seo"
  ...>
  ...
  <content-type name="news">
    ...
    <seo:enabled type="news" image-relation-type="photos"/>
    ...
    <ref-relation-type-group name="photo-relation"/>
  </content-type>
  ...
  <content-type name="video">
    ...
    <seo:enabled type="video" image-relation-type="photos"/>
    ...
    <ref-relation-type-group name="photo-relation"/>
  </content-type>
  ...
  <relation-type-group name="photo-relation">
    <relation-type name="photos">
      <ui:label>Photos</ui:label>
    </relation-type>
  </relation-type-group>
  ...
  <content-type name="photo">
    ...
    <seo:image representation-field="alternates" representation="SmallSquare" />
    ...
    <field name="alternates" type="basic" mime-type="application/json">
      <representations xmlns="http://xmlns.escenic.com/2009/representations"
      type="image-versions">
        ...
        <representation name="SmallSquare">
          <output width="100" height="100"/>

```

```

        <crop/>
        <resize/>
    </representation>
    ...
</representations>
</field>
...
</content-type>
...
</content-types>

```

Both of the `seo:enabled` elements in the above example include an `image-relation-type` element that references a `relation-type` element used for image relations. When creating a Google News sitemap entry for a `news` content item, the plug-in will include image data from the **first** relation it finds of the referenced type (`photos`, in this case). This means that the generated sitemaps will only contain information about one related image per content item.??

An image can have many different representations, so image content types also need to include an `seo:image` element that specifies the representation to be used when including image data in sitemap entries.

3.2 Editing the feature resource

The properties described below can be added to your publication's `feature` resource in order to control what information is included in Google News and Google Video sitemaps.

3.2.1 Google News-related properties

Required properties

If you have configured the Sitemap plug-in to use the Google News sitemap extension then you **must** include the following properties in your publication's `feature` resource:

`com.escenic.sitemap.publication.name`

This property must be set to the name of the publication **as it appears in Google News**.

`com.escenic.sitemap.publication.language`

This property must be set to the correct [ISO-639 code](#) for the language of your publication. You can use either the two letter or three letter codes (either `en` or `eng` is acceptable for English, for example).

Chinese is a special case: you must use either `zh-cn` for Simplified Chinese or `zh-tw` for Traditional Chinese.

Optional properties

If you have configured the Sitemap plug-in to use the Google News sitemap extension then the following property will be used if set:

`com.escenic.sitemap.keywords.tagStructure`

This property is **optional**. You can use it to specify which of your Escenic tag structures are to be used for including keywords in Google News sitemap entries.

Suppose for example that you have three tag structures: a geographical tag tree called **places.my.com, 2013**, a general tag tree called **topics.my.com, 2013** and a persons tag tree called **persons.my.com, 2013**. If you specify a **space delimited** list of tag structures as below:

```
com.escenic.sitemap.keywords.tagStructure=tag:places.my.com,2013
tag:persons.my.com,2013
```

then only tags belonging to **places.my.com, 2013** and **persons.my.com, 2013** will be included as keywords in the sitemap. A content item that is tagged with "Bergen" (from **places.my.com, 2013**) and "weather" (from **topics.my.com, 2013**) will appear in the sitemap with the keyword entry:

```
<news:keywords>Bergen</news:keywords>
```

The "weather" keyword is omitted because it does not belong to one of the specified tag structures.

If you do not specify this property then all tags associated with content items are included as keywords in the sitemap.

For detailed information about Google News extension elements, see <http://support.google.com/webmasters/bin/answer.py?hl=en&answer=74288&topic=10078&ctx=topic>.

3.2.2 Google Video-related properties

Optional properties

If you have configured the Sitemap plug-in to use the Google Video sitemap extension then the following property will be used if set:

com.escenic.sitemap.video.tagStructure

This property is **optional**. You can use it to specify which of your Escenic tag structures are to be used for including tags in Google Video sitemap entries.

Suppose for example that you have three tag structures: a geographical tag tree called **places.my.com, 2013**, a general tag tree called **topics.my.com, 2013** and a persons tag tree called **persons.my.com, 2013**. If you specify a **space delimited** list of tag structures as below::

```
com.escenic.sitemap.video.tagStructure=tag:places.my.com,2013
tag:persons.my.com,2013
```

then only tags belonging to **places.my.com, 2013** and **persons.my.com, 2013** will be included as tags in the sitemap. A content item that is tagged with "Bergen" (from **places.my.com, 2013**) and "weather" (from **topics.my.com, 2013**) will appear in the sitemap with a single tag entry:

```
<video:tag>Bergen</video:tag>
```

The "weather" tag is omitted because it does not belong to one of the specified tag structures.

If you do not specify this property then all tags associated with content items are included as tags in the sitemap.

For detailed information about Google Video extension elements, see <http://support.google.com/webmasters/bin/answer.py?hl=en&answer=80472>.

3.3 Editing SitemapConfig.properties

Two properties are set in this configuration file. If you need to modify the default settings, then you can do so as follows:

1. Copy **SitemapConfig.properties** to one of your configuration layers. To copy it to your common configuration layer, for example:

```
$ cp /opt/escenic/plugins/seo/misc/siteconfig/com/escenic/seo/
  SitemapConfig.properties \
    /etc/escenic/engine/common/com/escenic/seo/
```

2. Open **/etc/escenic/engine/common/com/escenic/seo/SitemapConfig.properties** for editing.
3. Make the changes you require and save the file.

The properties you can modify are:

entryPerSitemap

The maximum number of entries allowed in a sitemap. When this limit is reached, the Sitemap plugin splits the sitemap into multiple documents and adds references to all of them to the sitemap index document.

updateSitemapDuration

The maximum age of entries in update sitemaps, specified in hours. Content items that were published more than the specified number of hours ago will not be included in the sitemap.

sitemapBaseUrl

The url which will be used as the prefix of the sitemap url. For example if you set this as **http://example.com/** sitemap url for archive-sitemap will be - **http://example.com/sitemap/archive.xml** If this property is not set the publication url will be used by default.

3.4 Exposing The Sitemaps

In order for the sitemaps generated by the plug-in to be used, they must be exposed on your web site. To do this you need to add the plug-in's sitemap servlet and a corresponding servlet mapping to each of your publications' **web.xml** files. The entries you need to add are:

```
<servlet>
  <servlet-name>SitemapServlet</servlet-name>
  <servlet-class>com.escenic.seo.servlet.SitemapServlet</servlet-class>
</servlet>
...
<servlet-mapping>
  <servlet-name>SitemapServlet</servlet-name>
  <url-pattern>/sitemap/*</url-pattern>
</servlet-mapping>
```

These entries determine the sitemap index URLs that the plug-in's webapp will respond to. If your publication URL is **http://www.example.com/**, for example, then the sitemap index documents will be exposed in the following locations:

- Aggregated sitemap: **http://www.example.com/sitemap/archive.xml**
- Update sitemap: **http://www.example.com/sitemap/update.xml**

- Google News sitemap: `http://www.example.com/sitemap/googlenews.xml`
- Google Video sitemap: `http://www.example.com/sitemap/googlevideo.xml`

3.5 Configuring Section Parameters

If you want to exclude some sections of your publications from the sitemaps the plug-in generates, you can do so by adding this section parameter to the sections you want to exclude:

```
com.escenic.seo.sitemap.exclude=true
```

You **must** set `com.escenic.seo.sitemap.exclude=true` in Widget Framework config sections.

3.6 Configuring Solr

The Sitemap plug-in requires access to your installation's external solr instance (the one used by presentation hosts), and it expects to find it via the Tomcat environment variable `escenic/presentation-solr-base-uri`. Check your `/opt/tomcat/conf/context.xml` file. If it does not already contain an element like this:

```
<Environment
  name="escenic/presentation-solr-base-uri"
  value="http://localhost:8080/solr/"
  type="java.lang.String"
  override="false"/>
```

then you need to add it. Insert it as a child of the root `Context` element.

4 Administration

This section contains various items of administration-related information.

4.1 Sitemap Caching

The Sitemap plug-in's sitemaps are potentially risky targets for denial of service attacks, since they are dynamically generated. A continuous stream of requests to a sitemap will quickly result in an unmanageable load on the servers. It is therefore vitally important to place the sitemaps behind a cache server. Responses from the sitemap service include **Cache-Control** response headers containing **s-maxage** directives specifying how long the sitemaps are to be cached.

The default **Cache-Control** settings included in the headers are:

- Aggregated sitemap: **public, s-maxage=86400**
- Update sitemap: **public, s-maxage=300**

It is important to either ensure that your cache server is respecting these settings, or else set up similar caching rules manually using the cache server's administration interface.