



# **Roxen CMS 5.4**

## Blog Module Manual

 Roxen Internet Software AB  
© 2011 Roxen Internet Software AB.  
All rights reserved.

Under the copyright laws, this document may not be copied, in whole or in part, without the written consent of Roxen Internet Software.

Roxen Internet Software  
Box 449  
SE-581 05 Linköping  
Sweden  
[www.roxen.com](http://www.roxen.com)

Your rights to the software are governed by the accompanying software license agreement.

Every effort has been made to ensure that the information in this document is accurate. Roxen Internet Software is not responsible for printing or clerical errors.

Other company and product names mentioned herein are trademarks of their respective companies. Roxen Internet Software assumes no responsibility with regard to the performance or use of these products.

# Contents

---

<b>1</b>	<b>Blog Module</b>	<b>4</b>
<b>2</b>	<b>Administration</b>	<b>5</b>
2.1	Installation	5
2.2	Blog system	5
2.3	Customization	6
2.4	Component guide	6
2.4.1	blog-component	6
2.4.2	blog-utils-component	7
2.4.3	page-comment-component	8
<b>3</b>	<b>Operation</b>	<b>10</b>
3.1	Everyday use	10
3.2	Basic tasks	10
3.2.1	Login	10
3.2.2	Create new entry	10
3.2.3	Categorize blog pages	10
3.2.4	Add a category	11
3.2.5	Deleting a blog entry	11
3.3	Page comments	11
3.3.1	Comment on a page	11
3.3.2	Removing a comment	11
3.4	Advanced	11
3.4.1	Translate or change static strings	11
<b>4</b>	<b>Internal Tags</b>	<b>12</b>
4.1	<code>&lt;blog-count-pages&gt;&lt;/blog-count-pages&gt;</code>	12
4.2	<code>&lt;blog-strip-filename&gt;&lt;/blog-strip-filename&gt;</code>	12
4.3	<code>&lt;blog-suggest-dirname&gt;&lt;/blog-suggest-dirname&gt;</code>	12
4.4	<code>&lt;emit source="blog-page-categories"&gt;&lt;/emit&gt;</code>	12
4.5	<code>&lt;emit source="blog-page-components"&gt;&lt;/emit&gt;</code>	13
4.6	<code>&lt;emit source="blog-pages"&gt;&lt;/emit&gt;</code>	14

# 1 Blog Module

---

The Blog module consists of a series of components and stylesheets that makes it possible to become a part of the ever-growing blogging trend on the Internet using the Roxen platform.

## blog

### Fish

2008-01-18, 14:01 by [www](#) in [wildlife](#) [other](#)

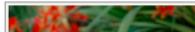
In non libero. Praesent leo massa, iaculis a, iaculis eu, egestas non, quam. Suspendisse vestibulum adipiscing ligula. Aliquam erat volutpat. Proin odio. Vestibulum ante ipsum primis in faucibus orci luctus et ultrices posuere cubilia Curae; Curabitur vel urna ac ligula venenatis consequat. Fusce nibh. Aenean elit. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Fusce eget eros ut mi varius pharetra. Phasellus ligula arcu, molestie sit amet, condimentum ac, dapibus sit amet, metus. Ut sodales scelerisque sem. Proin neque. Vivamus porttitor, nulla non interdum placerat, erat purus convallis lectus, vel feugiat arcu tellus ac lectus. Aliquam erat volutpat. Ut rutrum lorem eget est. Nunc tortor est, scelerisque sit amet, posuere sed, ornare ac, elit. Vestibulum lectus pede, nonummy et, adipiscing id, adipiscing et, dui. Phasellus ac massa.



[1 comments](#) [Leave a comment](#) [permalink](#)

### flowers

2008-01-18, 13:59 by [www](#) in [wildlife](#) [other](#)



Donec consectetur lacus at tortor. Pellentesque elit. Sed, nonummy et, adipiscing id, adipiscing et, dui. Phasellus ac massa.

### Categories

[landscape](#) (1)  
[other](#) (3)  
[travel](#) (1)  
[wildlife](#) (2)

### Latest comments

2008-01-18 14:09 av bob  
“ I can almost hear the Koalas! ”  
2008-01-18 14:06 av pablo  
“ In the shallowest waters... ”  
2008-01-18 14:05 av martin  
“ Those flowers are so lovely! ”



## Note!

The Category module is not bundled with the Blog module. The Category module is not required for running the Blog module, but is required to have categorized blog postings and categorized by-lines.

## 2 Administration

---

The following document will introduce the most common functionality of the Blog module and its components.

### 2.1 Installation

In order for the blog system to work properly you need to add a couple of modules to your site. Make sure that your license allows you to do so before you continue from this point.

#### 1. Add the following modules:

- *CMS Editor Component: Blog Component*
- *CMS Editor Component: Blog Utils Component*
- *CMS Editor Component: Blog Page Comments*
- *Blog module*

#### 2. Setup the module to use the correct path for blog entries.

This is done in the administration interface for your server > Sites > Your site > Modules > Blog module > Settings... Default is `/blog/`. You must also associate a database to your blog module. This is done by selecting the correct database under Blog database on the same settings panel as above.

#### 3. Create the corresponding directory on your site and a page containing the blog-component.

Make sure that you set the correct path to your blog-directory. Default is `/blog/`.

#### 4. Last steps:

- Visit the root folder of your site using the CMS and locate the file `/cms-components.xml`. Then bring up the Customize template...-page and make sure that the parameter `blog-stationery` is set to `/roxen-files/cms-sites/4.5/blog/cms-blog-stationery.xml`. The stationery controls which components that should show up on the blog pages when the user uses the automated process for creating blog pages. You can change this at a later stage by changing this parameter to point out a desired stationery file.
- While we are at it, also make sure that the parameter `blog-category-file` is set to `/blog-categories.xml` which is the default category file for the blog module. Please note that this file can be customized in the same way as the above mentioned stationery.

This should give you a page that presents all pages on the proper format in your blog directory.

### 2.2 Blog system

The blog system is built around the normal operation of Roxen CMS. The first page display uses a set of emits to deliver a view of a number of pages on the site. This allows the user to customize the view of both the individual blog pages and the first page view.

It is recommended to add the Blog Utils component variant 2 in order to create new blog pages. This causes a small admin panel to show up when in Edit mode. An editor can then log in to the site as usual and easily create new pages with this panel.

The package also contains a component for commenting pages on the site. This should be added to pages created with the control panel mentioned above.

Other variants of the Blog Utils component can be used to list information about categories, latest comments and a RSS feed with the latest pages on the blog.

## 2.3 Customization

The components have self-contained CSS rules so you can find the styles in their respective XSL file.

Either override these rules in a separate file by importing the desired self-contained CSS, e.g. by calling the `page-comment-component-comment-css` template for the comment component, then redefine the things you'd like to change and make sure that you use it instead of the rules in the file.

Another alternative would be to redefine the CSS rules directly in the containing file. It is also possible to use another method of your choice. Some users prefer to have a global CSS file which overrides a number of rules on the site.

## 2.4 Component guide

The following is a short guide to the components in the blog suite that can be used in the CMS editor.

### 2.4.1 **blog-component**

Provided by module: CMS: Blog Component

This is the main tag for the first page kind of view of the blog system. Use the tag where you want the blog to appear.

The component uses two attributes:

#### **entry-path**

Path to the directory where the blog pages are stored

#### **number-of-entries**

The number of entries to show on the first page view

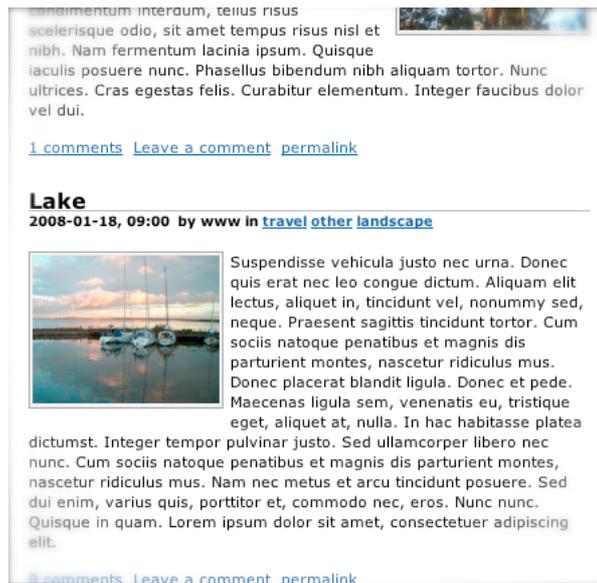


Figure 1. This is the result of a standard blog component implementation.

## 2.4.2 **blog-utils-component**

Provided by module: CMS: Blog Utils Component

This tag is used to extract some miscellaneous information from the blog module about the pages contained within the blog system. The information provided depends on the chosen variant.

- Variant 0: Lists all categories the number of pages within the respective category
- Variant 1: Lists the latest comments along with links and author names.
- Variant 2: Produces a small admin panel for creating new blog pages.
- Variant 3: Gives you a nice RSS-icon and enables the RSS-feed.

Except for the variants mentioned above, this component uses two attributes:

### **path**

The path attribute is used to point out different things depending on what variant of the component are being used:

- Variant 0: Path to the blog category file, default is `/blog-categories.xml`.
- Variant 1: Path to the blog directory, default is `/blog/`.
- Variant 2: Path to the blog directory, default is `/blog/`.
- Variant 3: Can be left blank.

### **count**

The count attribute is also used for different things depending on variant used:

- Variant 0: Can be left blank.
- Variant 1: Number of comments to show.
- Variant 2: Can be left blank.

- Variant 3: Can be left blank.

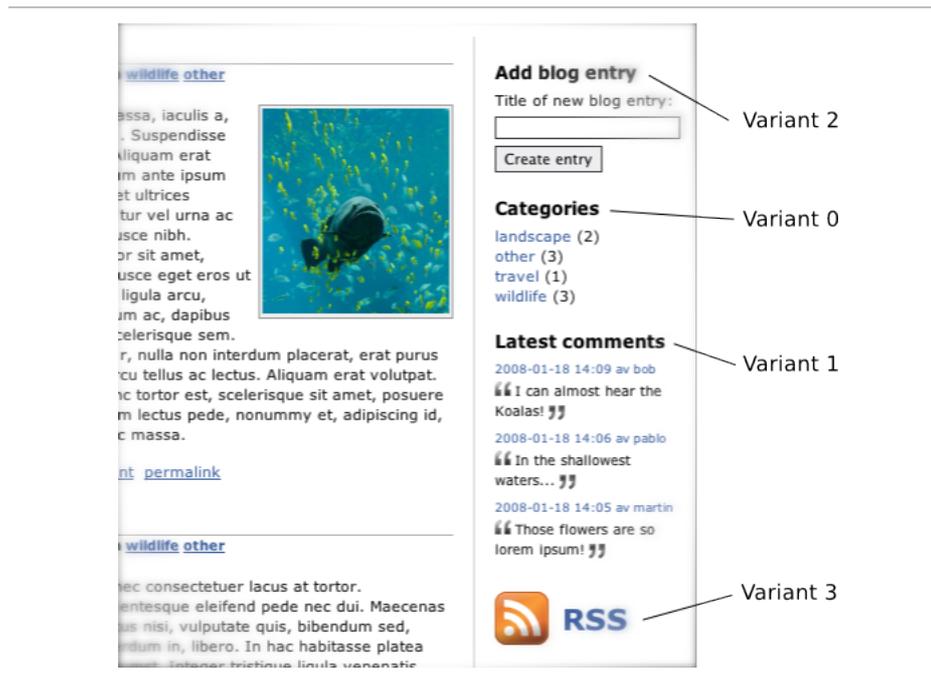


Figure 2. This is the result of standard usage of the `blog-utils-component` in a variety of variants.

The picture above is an example of how the utility components can be used on a page. This behaviour is accomplished by placing the utility components in the right column and the blog component in the left. Like so:

```
<div class="left-content">
  <blog-component>...</blog-component>
</div>
<div class="right-content">
  <blog-utils-component>
    ...
  </blog-utils-component>
</div>
```

This is a simplified example of how a blog first page might be constructed. The `blog-component` in `left-content` gives the blog overview and `blog-utils-components` provides the other functionality, such as category listing etcetera.

### 2.4.3 page-comment-component

Provided by module: *CMS: Blog Page Comment Component*

This is the tag you need to create the page comment form. The page comments are stored in the database associated to your blog module.

The component uses one attribute:

#### enabled

This can be set to enable or disable the component, i.e. you can choose to accept new comments or not.

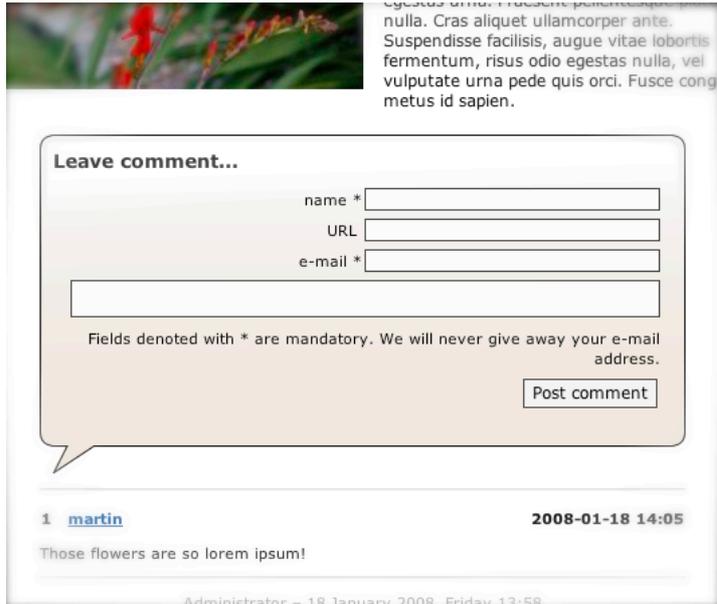


Figure 3. The standard appearance of the page comments component.

# 3 Operation

---

This section discusses the use of the blog module in everyday use. One could view this as an editors guide rather than a guide for the web developer.

## 3.1 Everyday use

As mentioned before, the blog module is based entirely on the Roxen CMS and relies on this to utilize all the blog functionalities. This means that no additional training is necessary for this module.

Once the system is set up it is easy to use and the intention is to limit the user modification needed.

## 3.2 Basic tasks

### 3.2.1 Login

Being based on the CMS the blog module uses the same system for user identification. Login to the site using one of the usual methods. For instance by pointing your favourite browser to `http://yoursite/edit/` where `yoursite` is the host name of your site. Enter user name and password.

This will log you in and complete the process.

### 3.2.2 Create new entry

To create a new page point the content editor to the location of the XML document where the Create new page-variant of the utility information component resides. Click the file and choose View from the appearing menu. This will show you the requested page. Input the desired title of the new blog entry in the text field and click Create page.

This will redirect your browser to the newly created page. This page should contain some basic components on which you can base the new entry.

Fill in the page with your text, add pictures and other content. Then hit the Save button on the top of the screen. Once you commit the changes the blog page will be registered, indexed and appear in the blog view.

#### Note!

The create component of the blog system automatically creates subfolders in where the blog pages are created. The default structure is

```
/blog/YYYY/MM/DD/your-blog-title/.
```

### 3.2.3 Categorize blog pages

This is done exactly in the same way as the categorization of other pages in the CMS. Use the content editor to browse through the file system and find your blog entry.

Click the icon marked Edit on the left hand side of the file view. Select Edit Category... from the menu. This will bring up the category tree with checkboxes on the side.

Check the categories you would like the file to adhere to and finish the operation by clicking the OK button. Changes are saved and indexed when you commit them to the server.

### 3.2.4 Add a category

To add a category, simply edit the contents of current blog category file (default is `/blog-categories.xml`) and press the node under which you'd like to add the new category and select **New Subcategory...** from the menu.

Follow the instructions in the wizard to complete the task.

### 3.2.5 Deleting a blog entry

Deleting an entry is again just as simple as deleting a page in the CMS. Login to the content editor and locate the blog entry you would like to remove. Delete the whole directory in which it resides. The changes will be visible to visitors when you commit them.

## 3.3 Page comments

### 3.3.1 Comment on a page

Commenting on pages does not exactly fit in to the normal work tasks of the editor, never the less it cannot hurt knowing how it's done. The whole process is very simple.

Try it out!

### 3.3.2 Removing a comment

Have you tried to add a comment to a page? Good!

Let's remove it!

This is almost as easy as creating a comment. Visit the page with the comment you would like to remove in edit mode, for example by adding `?__toolbar=1` to the URL. Hit Edit! The same effect can also be accomplished via the content editor.

You will now notice a link to the right of the comments marked delete. Simply click this link and the comment is gone.

## 3.4 Advanced

There is always time for some tweaks and tricks!

### 3.4.1 Translate or change static strings

The strings can be found in a file located in `/roxen-files/cms-sites/4.5/blog/blog-languages.xml` by default. Since the `/roxen-files/` directory is read-only you need to copy the contents you need to edit to a new folder. How this is done will not be covered here.

To add a new language to your blog, simply use the Edit icon and choose **Add Language....** Follow the instructions on screen to copy the contents of the existing language into the new one and then edit the file in your preferred text editor. You will find all static strings used in the blog system represented as variables in the file. Edit the strings to fit your needs and commit the changes. The blog pages can be translated using the same routines.

# 4 Internal Tags

---

Tags provided by the Blog module.

## 4.1 `<blog-count-pages></blog-count-pages>`

Provided by module: *CMS: Blog module*

Count the number of blog pages in a site.

Count the number of blog pages in each category:

```
<emit node='' source='category' file='/blog-categories.xml' category=''>
  &_.name;
  (<blog-count-pages language='&page.language;' category='&_.node;' />)
  <br/>
</emit>
```

Attributes:

### **category**

Count the number of blog pages in a specific category.

### **language**

Count blog pages in a language. This attribute has to be set if the site is a multi language site.

## 4.2 `<blog-strip-filename></blog-strip-filename>`

Provided by module: *CMS: Blog module*

This tag is used to strip the filename from a given path. It removes everything including the last slash in the full path.

Attributes:

### **path**

A full path of some sort, including filename first and foremost.

## 4.3 `<blog-suggest-dirname></blog-suggest-dirname>`

Provided by module: *CMS: Blog module*

This tag suggests a directory name when creating a new blog page.

Attributes:

### **title**

The title of the new blog entry. This is used by the tag to derive a suitable directory name.

## 4.4 `<emit source="blog-page-categories"></emit>`

Provided by module: *CMS: Blog module*

This tag emits the categories chosen for a blog page. It emits category ids from a CMS category file and should be used together with the emit category tags.

Emit the categories for the blog page defined by the RXML variable `blog-pages.path`:

```
<emit path='&blog-pages.path;' source='blog-page-categories'>
  <emit file='/&_.category-file;' node='&_.category-id;'
        source='category'>
    <a href='/?category=&_.node;'>&_.name;</a>
  </emit>
</emit>
```

Attributes:

#### **path**

The path of the blog page to emit categories for.

Emit entities:

#### **&\_.category-file;**

The Roxen CMS category file used.

#### **&\_.category-id;**

Category id for this category.

## 4.5 **<emit source="blog-page-components"></emit>**

Provided by module: *CMS: Blog module*

This tag emits CMS components from a blog page. An alternative to using this tag is to simply insert each blog page and apply an XSLT template in the insert tag.

Simple example of how to create a list of blog pages:

```
<emit source='blog-pages'>
  <emit path='&_.path;' source='blog-page-components'
        language='&page.language;'>
    <if variable='_.component-name = header-component'>
      <h2>&_.title;</h2>
    </if>
    <elseif variable='_.component-name = text-picture'>
      <p>&_.text:none;</p>
    </elseif>
  </emit>
</emit>
```

Attributes:

#### **path**

The path of the blog page to emit components for.

#### **language**

Emit a specific language. This attribute has to be set if the site is a multi language site.

Emit entities:

#### **&\_.component-tag;**

The value of the tag component-tag. I.e., if there is a text-tag in a component it will be accessible in the entity `blog-page-components.text`.

## 4.6 <emit source="blog-pages"></emit>

Provided by module: *CMS: Blog module*

This tag makes it possible to emit blog pages and information about blog pages on the site.

Emit all blog pages in a site and link to them:

```
<emit source='blog-pages'>
  <a href='&_.path;'>&_.title; &_.visible-from;</a><br/>
</emit>
```

Create a category selector and emit all blog pages of the selected category:

```
<emit source='category' file='/categories.xml' node='' category=''>
  <a href='/?category=&_.node;'>&_.name;</a><br/>
</emit>
<emit source='blog-pages' category='&form.category;'>
  <a href='&_.path;'>&_.title; &_.visible-from;</a><br/>
</emit>
```

Emit all Swedish blog pages in a site and link to them:

```
<emit source='blog-pages' language='sv'>
  <a href='&_.path;'>&_.title; &_.visible-from;</a><br/>
</emit>
```

Attributes:

### language

Emit a specific language. This attribute has to be set if the site is a multi language site.

### category

Emit only blog pages of a specific category. The category has to be specified in a Roxen CMS category file.

### path

Emit only the blog entry found in path. This is a glob pattern.

Emit entities:

### &\_.author;

Author id of blog entry.

### &\_.path;

Path of the blog entry.

### &\_.title;

Title of blog entry.

### &\_.visible-from;

The date the blog entry is visible from.