

UMBRACO I/S

# Blog 4 Umbraco

---

## Skinning Guide

pph

12/21/2009

Contains information on the css and html format for the blog for umbraco package and shows how to put together a simple skin package and redistribute it through [packages.umbraco.org](http://packages.umbraco.org)

## Contents

Introduction.....	3
The html.....	4
Where is the html located? .....	4
The overall html structure .....	4
Blog Post Html Structure .....	6
Blog Post Comments structure.....	7
Templates .....	8
The Template structure .....	8
Replaceable content areas .....	8
Creating a skin .....	9
Creating a skin manifest .....	9
Uploading the skin to the repository.....	9

## **Introduction**

The blog 4 umbraco package was done as a Christmas calendar in December 2009 by Tim Geysens, Niels Hartvig and Per Ploug Hansen.

It extend and reuses the blog package done by Tim Geysens, which reuses and alot of the work done even earlier by Niels Hartvig with the first blog package released in 2006. And includes a ton of xslt, produced by Morten Bock.

The goal for the blog package was to keep it so simple that it was easy to skin but still flexible enough to make some outstanding work, without having to tweak the template html.

For Blog 4 Umbraco, the term "skinning" only applies to applying another css file to the same html. So it doesn't include doing changes to the actual template html, only changing the appearance through css.

This document describes how the html is structured and how to package everything into a skin file that can be shared from the umbraco package repository.

## The html

The html for Blog 4 Umbraco comes from the Wordpress project "Sandbox" which is a project for css skinning a wordpress blog. It has been decided to reuse their schema as it has proven to be very flexible and easy to work with.

## Where is the html located?

The blog html is generated in 2 places:

- **The blog masterpage templates**, located in the settings section
- **The blog xslt files**, located in the developer section

Overall for the templates, the largest part of editable html is located in the BlogMaster template and the BlogPost template.

The BlogMaster contains all the html that are reused on all blog pages, as all blog templates inherit from it. Blogpost template contains some custom html which is duplicated in the BlogPosts.xslt file.

## The overall html structure

The html consist of very few main layout elements to keep track of. Everything is wrapped in a `<form>` element and a `<div id="wrapper">` element

For the blog header. 2 main elements are in use: `<div id="header">` and `<div id="access">`. These wrap around several items of child elements.

For the actual blog content, everything is wrapped in 2 elements. `<div id="container">` and `<div id="content">`. Inside these elements, blog posts and lists of posts will be generated.

For sidebar content. This is located in 2 elements, which both have the css class "sidebar". `<div id="primary" class="sidebar">` and `<div id="secondary" class="sidebar">` are sidebar elements which by default floats to the left and contains list of recent comments, tags, archives, etc.

Finally the page is closed off with a footer element which contains a bit of blog information. in the element `<div id="footer">`.

The simplified html layout elements are presented below. All child elements have been stripped from this view to just show how the main elements are structure.

```
<body class="umbraco blog">
  <form id="form1" runat="server">
    <div id="wrapper" class="hfeed">
      <div id="header"></div>
      <!-- #header -->
      <div id="access"></div>
      <!-- #access -->
      <div id="container">
        <div id="content"></div>
        <!-- #content -->
      </div>
      <!-- #container -->
      <div id="primary" class="sidebar">
        <ul class="xoxo">
          <li id="pages" />
        </ul>
      </div>
      <!-- #primary .sidebar -->
      <div id="secondary" class="sidebar">
        <ul class="xoxo">
          <li id="latest-comments" />
        </ul>
      </div>
      <!-- #secondary .sidebar -->
      <div id="footer"></div>
      <!-- #footer -->
    </div>
    <!-- #wrapper -->
  </form>
</body>
```

## Blog Post Html Structure

Each individual blog post is wrapped in a `<div class="hentry post" >` element. This wraps around the 3 elements which the blog post consists of: a `<h2 class="entry-title">` element which contains the title. A `<div class="entry-content">` element containing the body text, and finally a `<div class="entry-meta">` holding author, tag and publishing meta data.

The entire blog post html, which wraps around each individual blog post:

```
<div class="hentry post publish">
<h2 class="entry-title">Hello</h2>

<div class="entry-content">
  <p>Hello World body text</p>
</div>

<div class="entry-meta">
This entry was written by <span class="author vcard"><span class="fn
n">Administrator</span></span>,
posted on <abbr class="published">December 12th, 2009</abbr>

Bookmark the <a href="/" title="Permalink" rel="bookmark">permalink</a>.

Follow any comments here with the <a href="/">Comments RSS feed</a>.

You can <a class="comment-link" href="#respond" title="Post a comment">post a comment</a>.
</div>
</div>
```

## Blog Post Comments structure

Comments (if turned on) are arranged in a `<ol class="commentsList">` element, each comment have its own `<li>` and there is an alt css class on every other comment:

```
<div id="comments">
<h3>2 comments for "Hello world"</h3>
<ol class="commentlist">
  <li class="comment alt" id="comment-1">

    <div class="comment-author vcard">
      
      <span class="fn n"><a class="url url" href="#">Commenter</a></span></div>

    <div class="comment-meta">Posted 21. december 2009 at 11:23:50</div>

    <p>Hello</p>

  </li>

  <li class="comment alt" id="comment-2">
    <div class="comment-author vcard">
      
      <span class="fn n"><a class="url url" href="#">Commenter2</a></span></div>

    <div class="comment-meta">Posted 21. december 2009 at 11:26:52</div>

    <p>Hello again</p>

  </li>
</ol>
</div>
```

# Templates

## The Template structure

The blog package consist of multiple template files which are arranged in the following structure, for skinning these should not be changed.

- **BlogMaster**  
Contains the baseline html, such as <head> and <body> elements, as well as sidebar navigation, header and footer.
  - **Blog**  
The blog frontpage
  - **BlogPost**  
A single blog page with a post and comment list
  - **Blogpost Textpage**  
A simple textpage, does not contain comments or comment forms
- **CommentRSS**  
Generates comment RSS, so is not used for skinning.
- **RSS**  
Generates RSS for blog posts and tags, not used for skinning

## Replaceable content areas

The BlogMaster template contains several areas that can be inherited and replaced by child templates that inherit from the BlogMaster template (by default all blog templates installed inherit from BlogMaster)

- **header**  
Contains the Blog name and description
- **access**  
Contains the top navigation and the "skip to content" link
- **body**  
Where blog content and blog post lists are injected by default
- **sidebarPrimary**  
Sidebar area containing page navigation, categories and archives
- **sidebarSecondary**  
Sidebar are containing latest comments and blogroll
- **footer**  
Contains footer data

## Creating a skin

A skin in umbraco terms is nothing more than the css and images used by the css file. To turn this into a skin you can download from the skin browser, we need to put it into a special skin file, which is just a simple zip file with a specific file and folder structure.

The skin file must contain:

- thumbnail.png (a 100x100 px png image)
- preview.png (a 500xYYY px png image)
- skin.xml manifest
- /css folder
- /css/style.css stylesheet
- /css/images folder for any images used by the stylesheet

There is a sample skin file included with this document.

## Creating a skin manifest

The skin manifest is nothing but an xml file containing a bit of information about the skin.

```
<?xml version="1.0" encoding="utf-8"?>
<skin>
  <title>Runway 4 Blog 4 Umbraco (R4B4U)</title>
  <description>
    Sample theme for the blog 4 umbraco, reusing style elements from the runway package
  </description>
  <author>
    <name>Per Ploug Hansen</name>
    <url>http://umbraco.org</url>
  </author>
  <license>
    <name>MIT</name>
    <url>http://www.opensource.org/licenses/mit-license.php</url>
  </license>
</skin>
```

Edit the manifest and include it in the root of the zip file.

Change the preview and thumbnails to display the correct image

Copy the css file and it's images to the css folder. Make sure the css file is named style.css and that it references images in the /css/images folder relatively.

Finally, zip the file.

## Uploading the skin to the repository

Go to <http://packages.umbraco.org/skins/blog4umbraco/uploadskin> and upload the skin file.

It will then be publicly available as soon as it's verified.