BlogCFC

Welcome to BlogCFC, a ColdFusion based blogging application. BlogCFC is in use by over five hundred unique sites and has numerous features. If you run into problems with this blog, or have any ideas, please contact me. (Contact and support information below.) For the latest releases and information, please see http://blogcfc.riaforge.org and http://www.blogcfc.com.

License and Credits

Copyright 2006-2009 Raymond Camden

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

I request the following if you use my blog:

1) Please link back to my blog at http://www.coldfusionjedi.com.
2) If you find any bugs, or have any suggestions, please email me at ray@camdenfamily.com. You can also go to the BlogCFC Forums at http://www.coldfusionjedi.com/forums/forums.cfm?conferenceid=CBD210FD-AB88-8875-EBDE545BF7B67269. You may also go to the BlogCFC Project page at http://blogcfc.riaforge.org. Lastly – you can read news about BlogCFC at http://www.blogcfc.com.
3) If you *really* like the blog, you can always send me something from my wishlist, located here: http://www.amazon.com/o/registry/2TCL1D08EZEYE

Contributors

Many users have provided support, both in bug reports, suggestions, and even code, for this project. I cannot thank them all, but please note that I appreciate your work. BlogCFC would **not** be what it is today without the support of its community of users. I'd like to call out a few special contributors below.

BlogCFC makes use of the design/css skills of **Scott Stroz.**. His wishlist may be found here: http://www.amazon.com/o/registry/3OQX89CUTAV1X. Scott handled the "No more table" conversion of my layout for BlogCFC5, as well as providing design help in general and bug testing all around.

Jeff Coughlin and **Charlie Griefer** spent a lot of time and energy on the Related Entries feature. This was a significant new feature of BlogCFC5. Jeff's wishlist may be found here: http://www.amazon.com/o/registry/3MZKP36I6JCTL. Charlie's may be found here: http://www.amazon.com/gp/registry/registry.html/103-6263653-9438218?%5Fencoding=UTF8&type=wishlist&id=YLEZ85Q4LPCI

BlogCFC makes use of code from **Paul Hastings** (paul@sustainableGIS.com). Paul was a great help in adding localization to the blog. Many thanks go to him. We also make use of a custom tag (coloredCode) from Dain Anderson (webmaster@cfcomet.com). Many other people have helped with contributed bug fixes and ideas.

BlogCFC makes use of LylaCaptcha (http://lyla.maestropublishing.com/) by Peter J. Farrel. Please note that his code has a license as well. It may be found in org/captcha/lylaCaptchaLicense.txt. Brian Rinaldi (http://www.remotesynthesis.com/blog/) wrote an excellent article on integrating BlogCFC and Lyla. His article and code was used to add this feature to BlogCFC.

BloCFC makes use of SweetTweets (http://sweettweets.riaforge.org) from Adam Tuttle.

Deanna Schneider added Oracle and Query Username/Password support.

The BlogCFC logo was created by Joe Nicora.

BlogCFC makes use of CFFormProtect by Jacob Munson (http://cfformprotect.riaforge.org/)

Requirements

BlogCFC requires ColdFusion MX 6.1 or BlueDragon 6.2. Print support is only enabled in ColdFusion MX 7.

If you use the Developer Version of ColdFusion MX, a token is added to each request that will interfere with the RSS support as well as the Related Entries support in the administrator. The Developer edition should not be used for production sites. If testing BlogCFC on a Developer edition server, you will not be able to use the Related Entries feature.

Note – for MySQL, only the 4.1x release is supported. You **cannot** use the default MySQL 3 DSN type that ships with ColdFusion. To use MySQL, you must follow the directions here: http://www.adobe.com/go/6ef0253. If you are using ColdFusion 8, the newer MySQL drivers are provided for you. Just be sure to use MySQL4/5 for your DSN type.

Note - for Oracle, only version 10g is supported.

Access users should not use the "Access with Unicode" driver type. They should use the "Access" driver type.

BlueDragon Notes

In the latest version, there are a few issues with BlueDragon. First – Lyla CAPTCHA is not supported. You must disable CAPTHA support or the blog will not load in BlueDragon. (A future release will automatically disable it.) I already mentioned that print support is disabled, but this is done automatically by the blog engine.

Last but not least – I believe there is an issue with datasources that do not require usernames and passwords. ColdFusion MX allows you to pass a blank username and password while BlueDragon throws an error. For now I recommend updating your DSN to require a username and password.

Using OpenBlueDragon? The following note from **Scott Schug** describes how to enable SES URLs:

Apparently OpenBD can handle SES URLs, you
just have to turn on the appropriate filter for it:

http://tinyurl.com/9btf2w

Awesome! So I trotted over to my test server to try what Matt
suggested...and found to my dismay there *is* no web.xml out of the box
with the "Ready2Run" version of OpenBD. Grrr. After more searching and
gnashing of teeth, I wrote to Matt and explained what I was trying to do, and
the roadblock I'd hit. He didn't have any answers, but said he'd look into it.
Next day he forward me a long, multi-step post someone had made to the
OpenBD mailing list on the subject (attached). To make a long story short,
I realized what this guy was doing was teaching Jetty to handle the URL
rewriting, but it looked like only one simple step was necessary to turn on
the SearchEngineFriendlyURLFilter - add the following code to your
/jettyhome/etc/webdefault.xml file:

```
<filter>
   <filter-name>SearchEngineFriendlyURLFilter</filter-name>
   <display-name>SearchEngineFriendlyURLFilter</display-name>
   <description>SearchEngineFriendlyURLFilter</description>
   <filter-
class>com.newatlanta.filters.SearchEngineFriendlyURLFilter</filter-class>
   <init-param>
      <param-name>extensions</param-name>
      <param-value>cfm,cfml</param-value>
   </init-param>
</filter>
<filter-mapping>
   <filter-name>SearchEngineFriendlyURLFilter</filter-name>
   <url-pattern>/*</url-pattern>
</filter-mapping>
```

I dropped that in right at the end, before the final </web-app>, fired up
Jetty, went to my test version of BlogCFC, and it worked perfectly.

ColdFusion 6 Notes

Because of how CF 6 sets locales compared to CF 7, you should only use Java locales that can be mapped to valid CF 6 locales.

Oracle 10g Notes

The following notes come from Nick Hill. Thanks Nick!

When you install BlogCFC onto a server utilizing Oracle as your datasource, you will need to be aware of some factors that may inhibit the installation. The installation scripts are broken into 2 parts. The first is the table creation, grants and constraints. The second part is utilizing Oracle Synonyms to allow for general relation of tables names to limit the difference in the application code.

In some production scenarios access to Public Synonyms is blocked by the DBA. Oracle may be configured as a single database with multiple schemas. If public synonyms were allowed, any schema would be able to access the BlogCFC tables. If the previous scenario does not apply to your Oracle installation you may not have any issues with the Oracle scripts as they are currently written. If however your installation does match the above described scenario please be aware of the following changes to the scripts.

Oracle.sql
The first thing to be aware of is that when you are running this script you will want to be logged in as the SCHEMA OWNER for the Oracle schema and NOT the CF Datasource user that you are about to configure. It is a good practice to use a separate Oracle User (not the schema owner) for CFAdministrator to access your Schema. Specific to the script you will want to make changes to the lines that follow "insert into TBLUSERS"…

You will first want to change "&yourschema" to not be a schema name, but actually the separate Oracle User that you will assign to the datasource in CFAdministrator. Ex. (make sure you do not use these example names)

Oracle Schema: MyBlog
Oracle Schema Owner: BlogAdmin
CFAdministrator User: CFBlogAdmin

(This is the name you enter in CFAdministrator in the "Data & Services > Datasources > Oracle" Screen next to the "User name" label).

Before you run Oracle.sql replace "&yourschema" with the CFAdministrator User (CFBlogAdmin).
Logged in as the schema owner (BlogAdmin) you will run Oracle.sql. This will grant the necessary priviledges to CFAdmin.

Oracle_synonyms.sql
This is where Oracle is different from the other DB's. You will now want to LOGOUT as the Schema Owner (BlogAdmin) and log in as the CFAdmin User (CFBlogAdmin). Before you run oracle_synonyms.sql you will need to make some changes to the code. You will need to preface each tablename with the Oracle Schema that is being used. Ex.

CREATE SYNONYM TBLUSERS FOR MyBlog.TBLUSERS;

With all of the lines changed, now you will want to run this script as the CFAdmin user (CFBlogAdmin). What you have done is created a PRIVATE synonym for these tables that ONLY CFAdmin will be able to use which will ensure that you do not have to preface every table name in the Client code with additional scoping of the Oracle Schema.

Installation

When you unzip the installation package, you will have three main directories, client, org, and install. The client folder represents the main blog application. If your web site will only contain the blog, then obviously this folder will be your web root; you may rename this folder (be sure to update this change in Administrator Settings). The org folder should be placed under web root. This folder contains the CFCs used by the blog application. The install folder contains miscellaneous files to help you install the blog. This folder should be removed from your web root.

Next, decide which database you want to use. In the install folder you will find a mysql.sql file, a sqlserver.sql file, two Oracle files, and a MS Access database. These are the four database types supported by BlogCFC. Pick the one you plan on using it and set up a database manually, using your favorite db admin program. Set up a datasource pointing to your database. Open the blog.ini.cfm file from /org/camden/blog folder. Find the default section, dsn key, and edit the value. Set the dsn value to what you set in the ColdFusion Administrator. Set blogDBType to either MSSQL (for SQL Server), MYSQL (for MySQL), ORACLE (for Oracle), or MSACCESS (for Access).

Note – for MySQL, only the 4.1x release is supported. You **cannot** use the default MySQL 3 DSN type that ships with ColdFusion. To use MySQL, you must follow the directions here: http://www.adobe.com/go/6ef0253. I am not currently testing with MySQL 5, but it should work.

You also need to ensure blogdbtype matches your database. This is described in the **Blog Ini File** section.

Note: The blog.ini.cfm file is described in more detail later in this document. READ THIS: The blog.ini.cfm file is cached. This means if you visit the blog and make a change to the ini file, it will NOT be reflected. To refresh the blog's cache, visit your blog with ?reinit=1 at the end of the URL or use the Administrator.

Note to folks using non-Latin languages: Your DSN needs to be set up correctly. See this blog entry by Paul Hastings for more information:

http://www.sustainablegis.com/blog/cfg11n/index.cfm?mode=entry&entry=F9553D86-20ED-7DEE-2A913AFD8651643F

Also – for those using SQL Server – you may want to edit the collation settings, or remove them, in order for your database to work correctly. Those using SQL Server with an English blog should (probably) not need to worry.

You have two choices for placement of the org folder. If there will only be one blog on the box, you can simply place the org folder inside the client folder. If you will have multiple blogs, you should place the org folder in any location you want, but then set up a mapping called "org" that points to the folder you placed the org files. To repeat - no mapping is necessary if you place the org folder under the client folder.

BlogCFC makes use of SES (Search Engine Safe) URLs. You may have to update your ColdFusion install in order to make this work. Please see the Adobe tech note here: http://www.adobe.com/go/2addd247

At this point, your blog should actually be working, but obviously you will have no entries. In order to add entries, you should go to the BlogCFC administrator. If your blog URL is: http://localhost/blog, then the administrator would be http://localhost/blog/admin. You will be prompted to logon. Use **admin** for the username and **admin** for the password. You should change this password **immediately** or sometime soon. There is no interface to do this so you will have to manually edit the password in the database.

Unix users should make the /blog/images/captcha folder writeable by ColdFusion.

The Blog INI File

In the previous section, I described editing the Blog INI file (org/camden/blog/blog.ini.cfm) to set the DSN. There are other settings you should edit as well. You do not have to edit this by hand. There is an administrator tool for editing this file. Here is a typical set of entries:

```
[default]
dsn=blogdev
username=
password=
owneremail=ray@camdenfamily.com
blogURL=http://www.coldfusionjedi.com/index.cfm
blogTitle=BlogDev
blogDescription=The Dev Blog
blogDBType=MSSQL
locale=en_US
users=admin
commentsFrom=
mailserver=
mailusername=
mailpassword=
pingurls=
offset=-1
allowtrackbacks=1
trackbackspamlist=phentermine,casino
blogkeywords=
ipblocklist=
maxentries=10
usecaptcha=1
usecfp=1
allowgravatars=1
moderate=1
filebrowse=1
settings=1
imageroot=
itunesSubtitle=
itunesSummary=
itunesKeywords=
itunesAuthor=
itunesImage=
itunesExplicit=
usetweetbacks=
```

The section name, [default], refers to the name of the blog. If you have only one blog, leave this as default. If you have more than one blog on a server, you will want to copy the [Default] section and give it a new name, like [Other Blog]. You will then want to edit the Application.cfm file in the client folder and modify these lines:

```
<!--- Edit this line if you are not using a default blog --->
<cfset blogname = "Default">
```

The `dsn` obviously points to the datasource for your blog. BlogCFC supports multiple blogs in one DSN.

If your datasource requires a username and password, supply it in the `username` and `password` keys.

`owneremail` should be your email address. This is used when comments are posted to your blog.

`blogURL` should point to the URL of your blog. BlogURL **MUST** include the index.cfm file at the end. So if your blog is located at blog.org, do not simply use: http://www.blog.org. Instead, use the full URL: http://www.blog.org/index.cfm.

`blogTitle` and `blogDescription` and used in the RSS feed and in the main layout's top title bar.

`blogDBType` should either be MSSQL (for SQL Server), MYSQL (for MySQL), ORACLE (for Oracle), or MSACCESS (for Access).

`locale` is the locale setting for your blog. This is not a standard ColdFusion locale value, but a standard Java locale.

`users` are the users allowed to use the blog. If you do not specify a list of users, any user in the admin table can use the blog.

By default, comments are emailed out using a "from" address of the person who sent the email. If you set a value for `commentsFrom`, the email will instead be marked with that address. What that means is that when a user posts a comment to a blog entry, each email sent to the unique users of the thread will get an email from the address specified in that setting.

By default, the blog will send emails using the mail server specified in the ColdFusion Administrator. If you want to override that setting, use the `mailserver`, `mailusername`, and `mailpassword` settings. You do not need to specify the username and password if your mail server doesn't require authentication.

The `pingurls` setting lets you add ping support to your blog. This is described later in this document in the Ping section.

The `offset` property allows you to offset the posted dates of your blog entries. So if your blog server is in EST and you live in the CST timezone, you can use `offset=-1` to have the times of blog entries reflect your timezone.

The `allowtrackbacks` property should be 1 or 0 (or true or false) for allowing trackback support on your blog. Trackbacks are a way for people to link to your blog entries in a way so that they show up in your entries as well. Unfortunately, trackbacks tend to be spammed quite a bit. If you simply do not want to deal with trackback spam, just disable the feature.

Talking about spam – there are two ways to fight trackback spam in BlogCFC. You will get an email every time someone adds a new trackback. In that email will be a link you can click to delete the trackback. The "nuclear" option is the `trackbackspamlist`. If any word from this list appears in the trackback, the trackback entry will **not** be added to your blog. The person adding the trackback will **not** get a warning, but will simply not see their trackback show up. **Note: The trackbackspamlist also checks comments as well!**

The `blogkeywords` setting will be used in the meta tag of your blog. This should describe, in general, the content you plan on blogging about. It should be a list of keywords, separated by a comma. Spaces are ok. Here is an example: coldfusion, star wars, lost

The `ipblocklist` setting is a set of IPs that are prevented from adding comments or trackbacks to the blog. This could be used to block spammers or other people who may abuse your blog. You are also allowed to use wildcards. 192.168.1.* would block comments from anyone whose IP is in the range from 192.168.1.1 and higher.

The `maxentries` list controls how many entries are shown on your blog at one time.

The `usecaptcha` setting determines if CAPTCHA checking is turned on. If this setting is true, a reader will be forced to enter text from a CAPTCHA image before adding comments or sending blog entries.

Another method of spam protection is CFFormProtect. This is a spam blocker that checks multiple things that can flag a spammer. To enable this, set the `usecfp` value to 1. You can further confige cfformprotect by modifying the configuration information in the cfformprorect folder. Currently this is only used on the comment form, not the 'Send Blog' form.

Gravatars are a way to host images that represent a user's email address. If you turn on `allowgravatars`, BlogCFC will attempt to load a gravatar for each person's email address. If the person is not using Gravatar, a default image will be loaded instead.

The `moderate` setting determines if comment moderation is turned on.

The `filebrowse` setting determines if the file manager is available in the administrator. The file manager lets you upload files to your blog. You may want to disable this for clients who can't be trusted to not screw up their own file system.

The `settings` setting determines if the settings tool is available in the administrator. Like the filebrowse setting, you may want to turn this off if you think it will be too much for the client.

The `imageroot` setting helps determine where images are stored. These are images uploaded in the blog editor and slide show images. The default value is blank and results in images being uploaded directly in the relevant folders.

The `usetweetbacks` option determines if TweetBacks should be enabled for blog posts. TweetBacks are twits (from twitter.com) about your blog post. BlogCFC will make use of SweetTweets to search Twitter for short URL versions of your blog entry. This results in a slight delay on the first hit to your blog post, but the results will be cached for a few minutes.

The last batch of settings (the itunes group) all relate to podcasting. Please see the section on podcasting for more information.

Working with Your Blog

To begin working with your blog, you will want to enter the administrator (found at: yourblogurl/admin). In the left hand menu you will see various options:

- Home: Returns you to the administrator home page.
- Add Entry: Takes you straight to the form for adding a new entry.
- Entries: Lets you browse, add, delete, and edit blog entries.
- Categories: Lets you browse, add, delete, and edit blog categories.
- Comments: Lets you browse, add, delete, and edit blog comments. This is a useful place to remove spam comments.
- Moderate Comments: Used to moderate comments.
- Pages: Lets you browse, add, delete, and edit blog pages.
- Textblocks: Lets you browse, add, delete, and edit textblocks.
- Refresh Blog Cache: Refreshes the blog cache.
- Settings: Lets you modify the blog settings defined in the INI file.
- Subscribers: Lets you browse and remove subscribers.
- Mail Subscribers: Lets you send an email to your blog subscribers.
- Trackbacks: Lets you browse, add, delete, and edit trackbacks.
- Your Blog/Your Blog (New Window): Simply sends you to the public facing part of your web site.
- Your Blog Stats: Takes you to your blog stats. This is a public page.
- File Manager: Lets you upload and view files.
- Slideshows: Lets you work with slide shows.
- Update Password: I bet you can guess what this does.
- Users: Let's you modify users.

You will want to start by creating an initial blog category. You must have at least one blog category before writing an entry. After you have created a category, you may start writing blog entries.

Blog Editing Options

When working with blog entries, there are a few special tags and features that you can use to enhance your entries:

: Sometimes a blog entry may be very long. It is preferable to show only a portion of the entry on the main blog and have the detailed link contain the entire entry. By using the tag in your blog entry, any text after the tag will only show up on the individual entry display. You **must** include the closing / in the tag. When blog entries are emailed, only the text before the tag is included.

<code>: The code tag allows you to wrap sets of code that you want to show up in your entry. For example:

```
This is a sample entry. Blah blah blah. Here is some code…

<code>
<cfif thisTag.exectutionMode is "end">
     <cfexit>
</cfif>
</code>

This is more text in the entry.
```

Any code inside the <code> block will be escaped (< and > tags changed to < and >), color coded, and will have line breaks changed to <br> tags.

<textblock label="">: BlogCFC 5.1 added textblock support. A textblock is simply a block of text with a label. You can include textblocks dynamically in your blog entries by using the <textblock> tag:

```
Here is my standard footer: <textblock label="footer">
```

Entries can have "enclosures", which are simply attached files. This is a handy way to attach a resource to a blog entry. One of the features of BlogCFC is that if you attach an image, the file will also be included at the top of the blog entry. This is a handy and simple way to display an image with a entry.

Scheduled Releases

BlogCFC supports delayed entries. This means you can write an entry now that won't be displayed until later in the day. Subscribers will not get an email on the entry until the entry is published. BlogCFC will automatically add a scheduled event to handle the entry release.

Draft Mode

BlogCFC technically doesn't support a draft mode, but an entry can be saved with the Released setting to false. This means that the entry will not be shown on the blog. If you want to write an entry and save it without publishing it, simply set the Released value to false.

HTML/XML-RPC Editors

BlogCFC uses a simple text area to handle entry editing. If you want fancier editing, you have two options. One is to simply use an XML-RPC capable blog editor. Your other option is to use an HTML edit control like tinyMCE. I've made this easier by abstracting the textarea field out of the blog editor. You can find this in the tags folder. The filename is textarea.cfm.

For an example of how to use tinyMCE, see this blog entry: http://www.succor.co.uk/index.cfm/2006/10/10/Adding-TinyMCE-to-blogCFC-admin-area

Handling Users and Security

BlogCFC 5.9.4 added new features in regards to blog users and security. You can now add, edit, and delete blog users. (Note – you should not delete users who have written articles.) For each user you edit you can now specify roles. A special role, "Admin" allows for everything. You can supply specific permissions though if you want to prevent users from changing categories for example. BlogCFC 5.9.4 ships with a small set of permissions but these will be expanded in the future.

Working with Pages and Textblocks

The 5.1 version of BlogCFC added basic CMS (content management system) capabilities to the blogging package. While the basic blog application works great for diary style entries, there was no way to create basic content outside of the blog. BlogCFC 5.1 changes that with the pages and textblock options.

Pages

Pages are exactly what they sound like - complete pages of content. So imagine you wanted to add a "About Me" page to your blog. By using the pages feature, you could create a page that describes you. Once you give it a title, BlogCFC will automatically creates a SES (search engine safe URL) for the page and display it in the administrator. All pages live off the page.cfm file. So your "About Me" page might have this url:

```
http://www.yourblog.com/page.cfm/About-Me
```

You can add any number of pages you want. To link to a blog page, simply use the URL displayed in the administrator. Pages follow the same editing rules that blog entries do. This means you can include code and textblocks directly in the content. The only tag not supported is the <more/> tag.

Textblocks

A textblock acts much like a page. However, these are used for smaller pieces of content. For example, imagine you have a pod with your favorite links. Normally to update this you would need to edit the file and FTP it to the server. Textblocks lets you store the link pod data in the database. To use textblocks, simply use the textBlock application control:

```
<cfoutput>
#application.textblock.getTextblockContent("links")#
</cfoutput>
```

Textblocks are identified by their labels so ensure that you do not duplicate two textblock labels. You can include the content of a textblock inside a blog entry by using the <textblock> tag. Please see the blog editing section for more information.

Working with Slide Shows

BlogCFC 5.2 added a new feature: Slide Shows. There are two ways to create slide shows. One way to create a slide show is to simply upload a folder of images underneath your blog's images/slideshows folder. If your folder was named "foo", you have created a slide show named foo. To view the slide show, simply go to:

```
http://www.yourblog.com/slideshow.cfm/foo
```

BlogCFC will automatically discover all GIF and JPG files in the folder and will create a slide show based on those images. You should do your best to size your images beforehand as BlogCFC will not do any automatic sizing.

The second way to create slide shows is via the administrator. The administrator lets you add, edit, and delete slide shows. It also lets you add titles to your shows as well as captions to your pictures.

Podcasting

Podcasting support was added to BlogCFC 5.9.1. The major work for this was provided by Brian Meloche. Podcasting is a large topic so I'll just cover how BlogCFC handles it. I assume you've got your MP3 files already prepared and ready to go.

The first thing you need to do is add the MP3 to the blog entry. You can do this either by uploading the MP3 when editing the blog entry, or by using another program (like an FTP client) to upload the MP3 file. You can then manually type in the name of the file in the blog entry.

As soon as you have used an MP3 file for a blog entry, a simple, Flash based audio player will be rendered with the blog entry. Even if you have no plans of being a podcaster, this could be useful for folks who want to attach songs or other sounds to a blog entry.

BlogCFC now supports new settings for podcasting. They are: itunesSubtitle, itunesSummary, itunesKeywords, itunesAuthor, itunesImage, and itunesExplicit. These settings are used in the RSS feed and are used by various feed readers to help flag blogs with podcasts attached to them.

Miscellaneous

Layout Options

The look and feel of your blog is controlled by files in the client directory. The main file you will want to examine is /tags/layout.cfm. This, along with /includes/styles.css and /includes/layout.css, control most of the look and feel of the blog. Other files you may want to look at: index.cfm controls the display of entries. If you wanted to modify how an entry is shown, you would modify this file. The file /tags/podlayout.cfm controls the display of pods (right hand content). Do not forget that caching is used to speed up the display of the blog. If you modify the layout, hit your blog with ?reinit=1 in the URL to force a refresh of the cache.

The installation folder contins a "skins" folder that has a few other files you can use to change the layout of your blog. There are four options: Fixed versus Liquid (liquid means the content stretches with the window) and an option to place the pods on the left.

Comment Moderation

You can enable comment moderation by editing the blog.ini.cfm file or by using the Admin settings page. If comment moderation is enabled, comments will not show up until you approve them. You can use the footer of the comment email to approve comments with one click.

Pods

"Pods" is simply my term for the blocks of content in the right hand column. These tags are loaded by the /tags/layout.cfm file. To ensure a common look and feel, each pod file should use the /tags/podlayout.cfm file. This controls the size and tabular look of the pods. There are multiple pods included in the zip. You can modify/add/delete these to suit your tastes.

Please note that some pods included with BlogCFC are not actually used out of the box. I encourage you to take a look at the pods and see which ones you would like to use.

Caching

The blog makes use of scopeCache, a custom tag I developed to enable easy RAM caching. You should be aware, however, that this caching means changes to layout may not be reflected immediately. You can either disable the tag in the index.cfm file, or you can simply refresh the cache using the Administrator. Caching will be automatically refreshed whenever you add, edit, or delete an entry. Basically, you need not worry about it unless you are working on the layout.

Security Options and Multi-User Blogs

Blog 3.5 added support for multiple blogs and users per dsn. To add, edit, or delete users, you must edit directly in the database itself. Any user defined in the tblUsers table will have access to all the blogs that exist in the database. You can use the blog.ini.cfm file to limit users to particular blogs. Please see the Blog INI section for more information.

Internationalization

BlogCFC makes use of a resource bundle CFC developed by Paul Hastings (and with slight modifications by myself). All strings (things like button labels, etc) are controlled by a properties file in the includes directory. By default, the blog uses main_en_US.properties. If you specify another locale in the blog.ini.cfm file, then the Blog will attempt to load that particular resource bundle. To create a new resource bundle in your language, copy the en_US file and make sure you translate each and every setting.

Comment Notification

By default, when a user posts a comment to a thread, the administrator for the blog will get an email. Users also have the option to subscribe to a thread. When they do, they will automatically get notice when another user posts to the thread. Users have two options to unsubscribe. If they post again, and do not check the "Subscribe" checkbox, they will be removed from the thread, even though they subscribed earlier. The email sent out also contains a link to unsubscribe. If you want to modify how comment notifications look, edit addcomment.cfm. You will see where the email is generated. Everything there is pretty straightforward except for the %unsubscribe% token. This is a special value that gets replaced with the unsubscribe URL and values specific to the person getting the email. Note that as the owner of the blog, you get emails for all postings. The unsubscribe link will be marked as unavailable.

By default, all emails sent out will be marked with a footer that identifies the BlogCFC application and gives credit to Raymond Camden. You have the right to remove this footer, or modify it as you see fit.

Using Aliases

BlogCFC makes use of UUIDs to identify blog entries. Some folks do not care for these URLs. A feature of BlogCFC is the ability to give an alias to an entry. So for example, a blog entry on Star Wars could have an alias of "starwars." To load an entry with it's alias, you must use this syntax:

```
index.cfm?mode=alias&alias=starwars
```

SES URLs and Aliases

BlogCFC version 3.8 added support for SES URLs. So, a typical URL could look like so:

http://host/client/index.cfm/2005/7/12/Testing-my-new-title-maker

By default, when you make a new blog entry, an alias using the year, month, date, and title will be created. You do not need to worry about this typically. The only potential problem will be if two (or more) entries in the same day have the exact same title, or a title similar enough so that they create the same shortened version.

If you have problems with the SES URLs, please see this technote: http://www.adobe.com/go/2addd247. Also ensure your blogURL property has "index.cfm" in the value.

Ping

Some blog aggregators support 'ping', which simply means a way to tell the aggregator that you have updated your blog. This is typically done by simply loading a URL. For aggregators that support this feature, you can ping them from your blog when you add a new entry. Simply add the ping URL to the pingurls setting in the blog.ini.cfm file. You can supply multiple ping URLs by simply separating them with a comma.

Note - BlogCFC supports Technorati pings. In order to ping Technorati, simply add @technorati to your blog.ini.cfm file in the pingurls setting. This can be by itself, or with any other ping URL.

BlogCFC also supports the Weblogs API (http://www.weblogs.com/api.html). To ping weblogs.com, add @weblogs to your ping list.

BlogCFC also supports Icerocket. To ping them, add @icerocket to your list of pingurls.

RSS

BlogCFC makes use of RSS 2.0. You can offer RSS 1.0 support by simply adding version=1 to the RSS URL. So if your normal RSS URL is:
http://www.coldfusionjedi.com/rss.cfm

You would use this to offer RSS 1.0:
http://www.coldfusionjedi.com/rss.cfm?version=1

It is certainly ok to offer both RSS feeds to your visitors, but only RSS 2.0 will support enclosures.

You can create "RSS" blends of multiple categories. This is done by first getting the ID values of each category. You can find these in the BlogCFC admin. Once you have the IDs, you append them to the end of the core RSS url. If your blog was located at blogcfc.localhost.com, your RSS url would look like so:

http://blogcfc5.localhost.com/client/rss.cfm?mode=full&mode2=cat&catid=96F8AE90-A8BC-1230-CC4F71ED36CF7E51,96F8DD56-9C3A-BEF5-88133F51DC54E2FB

You can put any number of category IDs in this list. (Up to the maximum length of a URL.) This RSS "blend" will show the last ten articles from any of the categories listed.

Enclosures

BlogCFC 3.9 added support for enclosures. These are media files associated with a blog entry. This allows support for pod casting. While you can attach any file to a blog entry, normally you will want to attach an MP3 file. Once a file is attached, it will show up in the entry as a download, and be displayed in the RSS 2.0 feed. If you enclose an image, it will also be automatically added to the blog entry in the top left corner.

You also have the option of uploading an enclosure manually (via FTP for example) and setting an enclosure file name directly in the blog entry editor. This is handy for files that are too large to upload via HTTP.

Subscribers

BlogCFC allows folks to subscribe to your blog. When a person subscribes, they will receive each and every posting you write. They will not receive an email if you simply edit an existing post.

Print Format

BlogCFC offers a print format version of blog entries. This is only supported on ColdFusion MX 7. If you are not running BlogCFC on ColdFusion MX 7, you will not see the Print link.

Google Site Maps

BlogCFC now supports Google Site Maps. Google Site Maps are a way to let Google know the 'map' of your site. For BlogCFC, this is simply a listing of all the blog entries. More information may be found here:

https://www.google.com/webmasters/sitemaps/docs/en/about.html

BlogCFC uses the file, googlesitemap.cfm, in the root of the client folder to enable site map support. Note - I have problems using a CFM file with Google. I found that Google would only accept my file when I added a URL parameter at the end to 'fake' an XML file. In other words, I provided a URL like so:

http://www.foo.com/googlesitemap.cfm?ext=xml

This URL variable was ignored by the code, and seemed to please Google. As you know, we must all please Google!

Dynamic BlogCFC Instances

BlogCFC supports dynamic blog instances. This allows for Blogger.com type sites. While I do not have 'proper' documentation right now, you can view a blog entry on the topic here: http://www.coldfusionjedi.com/index.cfm/2006/10/7/Dynamic-BlogCFC-Instances

"Blended" Category Pages

In the RSS section I talked about how you can manually create RSS links for multiple categories. The same can be done for the front end. As with the RSS support, this is done by finding the category IDs and adding them to the URL, like so:

http://blogcfc5.localhost.com/client/index.cfm?mode=cat&catid=96F8AE90-A8BC-1230-CC4F71ED36CF7E51,96F8DD56-9C3A-BEF5-88133F51DC54E2FB

Note that you cannot use the category aliases. You must use the IDs.