

Building a ColdFusion Server Behavior in UltraDev

By Tom Muck



<http://www.basic-ultradev.com>

For you ColdFusion users who have dealt with the <CFMAIL> tag, you likely already know that either HTML-based emails or text-based emails can be sent from the tag by setting the Type attribute. What many of you may not know is that you can send multipart text/html emails from <CFMAIL> by inserting the correct headers into the email. I'll show you how this is done, and in the process introduce a great feature of Macromedia's Dreamweaver UltraDev -- the Server Behavior Builder-- that will create Server Behaviors that automatically insert the customized code with all of your parameters filled in.

Before I getting to that, let me provide a brief introduction to UltraDev, which will include an introduction to the concept of Server Behaviors.

Macromedia ColdFusion

For ColdFusion users everywhere, the development environment of choice is ColdFusion Studio. ColdFusion Studio has everything a ColdFusion developer needs to be able to write ColdFusion code and develop Web applications. With the release of UltraDev – and the merger of Allaire with Macromedia – it's time to rethink your development strategy. Not because ColdFusion Studio is going anywhere, but because using the two programs together can significantly increase your workflow and decrease production time.

UltraDev is a program built upon the framework of Macromedia's highly successful Dreamweaver program. Dreamweaver has been at the top of the heap for Web designers for years, but UltraDev finally brings a programmer's perspective to a designer's application. Inside UltraDev, the seemingly contrasting roles of designer and programmer are made more complementary. This is one of the breakthrough concepts behind UltraDev that puts it at the forefront of Web application development tools on the market today.

Server Behaviors

A couple of the great timesaving features of UltraDev are the built-in Server Behaviors and Live Objects. With these features, you can whip up quick prototypes of sites, data access pages, or site administration sections. Dreamweaver has always used the term Behaviors to describe the client-side scripts that perform some action in your page, usually in an event of an HTML tag. Typically in Dreamweaver you would insert an object on the page, such as a form field, and then apply a Behavior to an event of that field or the form itself. Behaviors include things like client-side validation, rollovers, and show/hide layers.

A Server Behavior, on the other hand, is the UltraDev term for server-side script that performs some action. Some of the built-in UltraDev Server Behaviors include the user authentication Server Behaviors, which will check a database for a duplicate username and allow a person to log in to a site by setting a session variable, and restrict access to pages based on access levels that you can set up on a site-by-site basis.

After you've applied a Server Behavior to the page, you can focus on adding functionality rather than the repetitive task of rewriting the same action. How many times have you hand-coded a table lately? It's certainly much easier to tell your program to

insert a 2 row, 6 column table than to hand code the <table>, <tr>, and <td> tags. Server Behaviors operate under the same principles. You instruct UltraDev to apply an Insert Server Behavior to your form with all the database interaction on the page, and it does it.

Server Behaviors vs. Snippets

Server Behaviors are simply pre-written sections of code. In ColdFusion Studio, these would be the equivalent of snippets (Figure 1). ColdFusion snippets are easily added to ColdFusion Studio and easily inserted into your page by dragging into the document from the Snippets tab. This certainly speeds the development time.

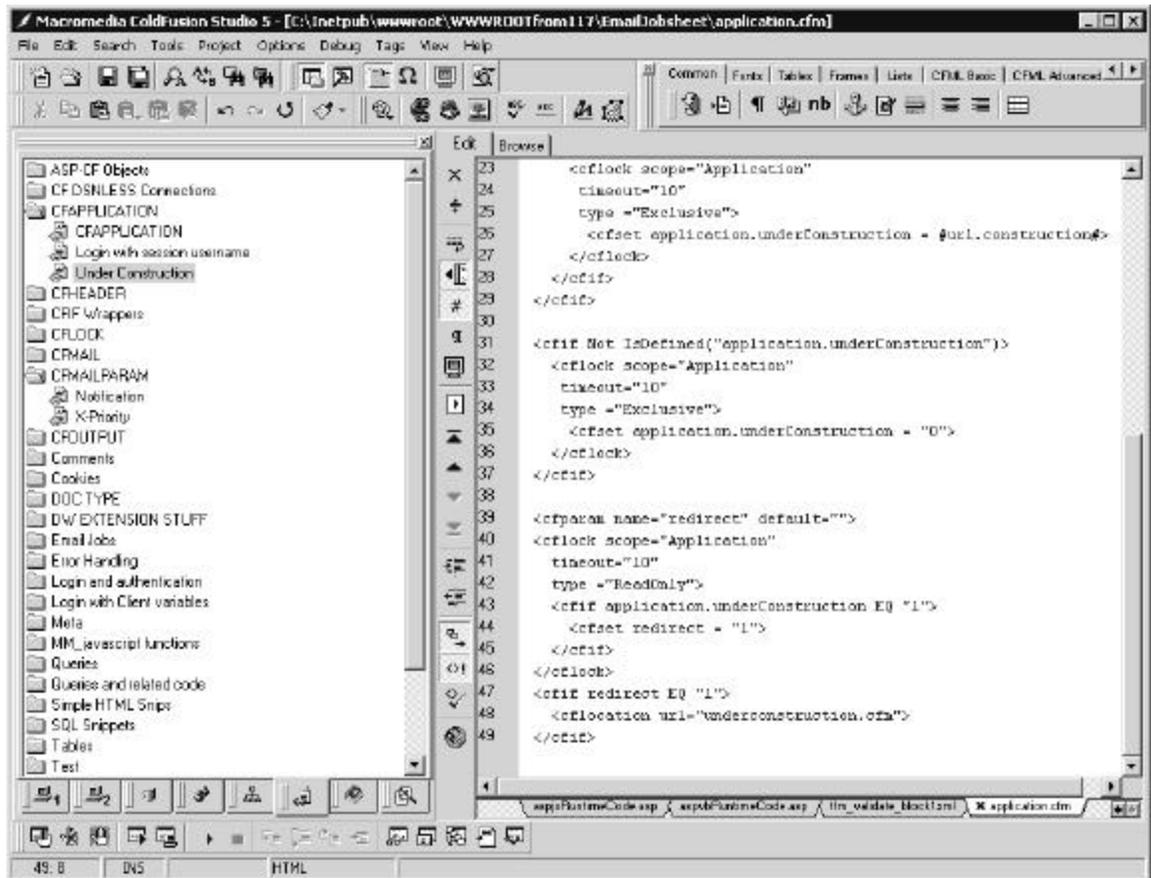


Figure 1: Using Snippets in ColdFusion Studio

UltraDev takes the concept of the snippet to another level with Server Behaviors. If the code has parameters that can change, the Server Behavior will have an interface that will allow you to insert the parameters. For example, a Repeat Region Server Behavior inserts a <CFOUTPUT QUERY="myquery"> on your page around the selected code. Typically you would apply the Repeat Region to a table row to display a dynamic table. The parameters that are user-defined are for recordset (<CFQUERY>) and number of rows. If you choose to display a limited number of rows, there are other Server

Behaviors that allow you to page through the resultset of the query. All of the UltraDev Server Behaviors work together, which makes them quite convenient.

One of the arguments by CF programmers against UltraDev is that they don't like the code that it generates. UltraDev uses a lot of CFScript, and CF guys are more comfortable with CF tags. Luckily for them, Server Behaviors are almost as easy to add to UltraDev as snippets are to CF Studio with the use of UltraDev's built-in Server Behavior Builder (SBB). All you have to know is the code that you want to insert, and where you want to insert it. The SBB creates a user-interface that will allow you to easily type in the parameters for your code when you apply the SB to the page.

Before moving to the multipart email Server Behavior, here's how you would create a simple Server Behavior with the SBB that inserts a test for a Macintosh user and redirects him to a different page. This SB uses one block of code and one user-defined parameter (the page to redirect to). The code is as follows:

```
<CFIF FindNoCase( 'mac' , #CGI.HTTP_USER_AGENT# ) >  
  <CFLOCATION URL="macpage.cfm" ADDTOKEN="yes" >  
</CFIF>
```

If this were a snippet in CF Studio, you could type in the URL parameter each time you wanted to use it. To turn this snippet into an UltraDev Server Behavior, you need to follow these steps (assuming you are already in a ColdFusion site in UltraDev):

1. Open the Server Behaviors palette, click the plus sign (+) and click New Server Behavior (Figure 2).

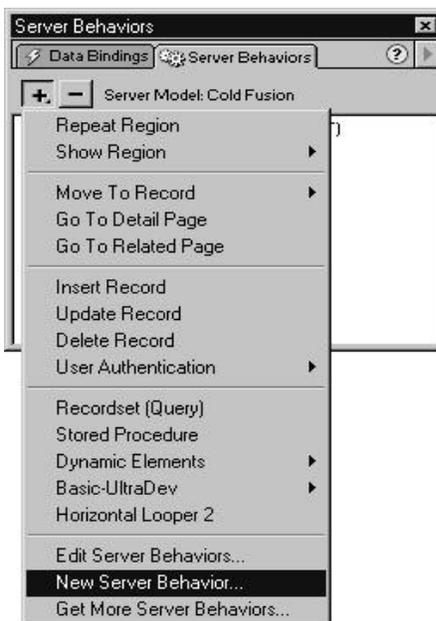


Figure 2: Adding a New Server Behavior

2. Give the new Server Behavior a name, such as "Redirect User if Macintosh" and click OK (Figure 3).

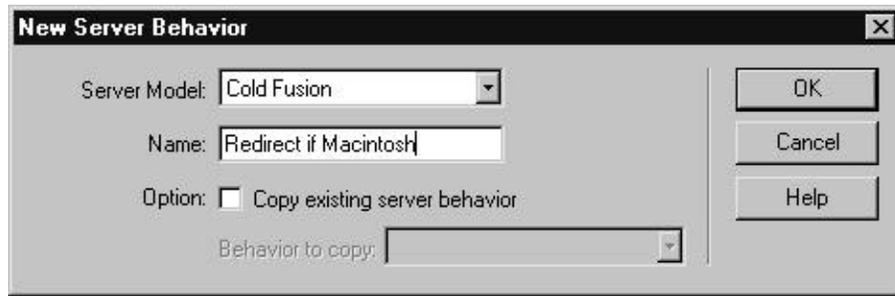


Figure 3: Giving the new Server Behavior a name

3. Click the plus sign (+) next to the title Code Blocks to Insert. This will bring up a dialog box to allow you to name the code block. You can click OK to keep the default name that is given to the block by the Server Behavior Builder.
4. Copy and paste the code block into the text area on the SBB (Figure 4).

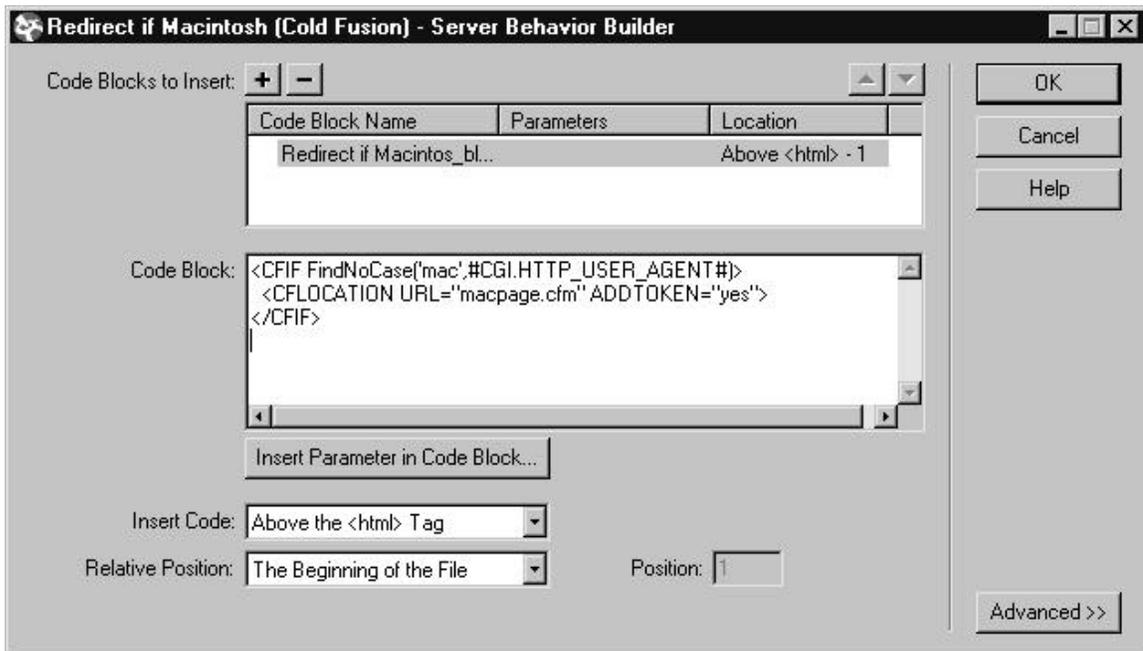


Figure 4: Adding a code block to the new Server Behavior

5. Here is where the fun begins. Highlight the URL attribute within the quotes -- this is the attribute that you are going to parameterize. Click the Insert Parameter In Code Block button and give the parameter a name, like Page To Redirect To. This name will show up in the interface of the Server Behavior. If you wanted to parameterize the ADDTOKEN attribute, you could do that also.
6. Next you'll have to choose a location where you want to place the code. Unlike snippets in CF Studio, UltraDev Server Behaviors have many possible locations. You'll begin to appreciate this when you have more than one code block to insert

in different places. In this case, choose to Insert Code Above the <html> Tag and the Relative Position attribute should be set to The Beginning of the File. Click Next. That brings up the dialog box shown in Figure 5.

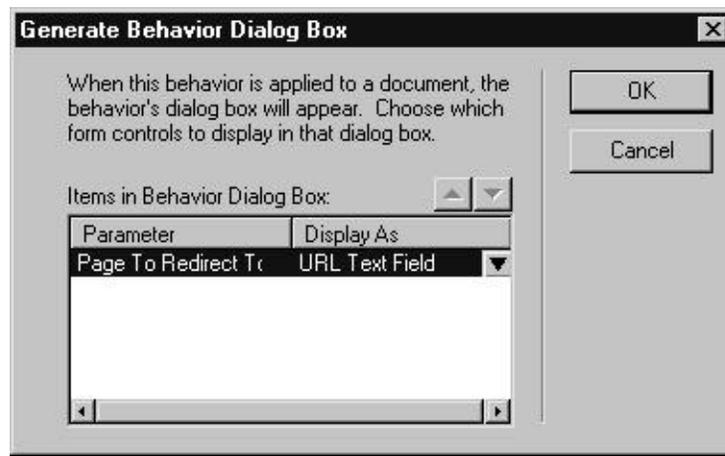


Figure 5: This dialog box allows you to choose the input type for each user-defined parameter.

7. The next step is to choose the type of field that you will use to input the parameter in the code. There are several types available by default, including a basic textfield. For this SB, the URL Text Field is the best choice. This will insert a text field with a browse button next to it.
8. Click OK and you're done. The Server Behavior will now appear in the Server Behaviors palette and you can apply it to any page with a click. You may notice that the Browse button allows you to pick a file from the file system and will automatically insert the correct relative path to the file.
9. With a simple Server Behavior under your belt now, you should be ready to take on the challenge of the Multipart CFMAIL Server Behavior.

Creating the Multipart CFMAIL Server Behavior

The code in Listing 1 shows the content of the Multipart CFMail Server Behavior. When it's complete, there will be nine parameters that will need to be defined by you when the SB gets applied to a page. The parameters are plainly labeled in the code listing by using the UltraDev Server Behavior format of @@parameter@@. These parameters will appear in the Server Behavior interface.

The <CFMAIL> tag is fairly limiting for emails, but you can push the limits of it by defining your own multipart header and boundaries. Since spacing and formatting of the headers and boundaries are critical (and spell the difference between a successful email and indecipherable text) this code makes a perfect candidate for a Server Behavior. All you have to do is fill out a few parameters and drop the code into the page.

One of the parameters that will be included in this Server Behavior is the Return-path parameter. This parameter is utilized in a situation where you are sending an email

for a client and you don't want the bad emails to be returned to the client--you can track these yourself by including a Return-path. This parameter can't be specified in the <CFMAIL> tag. It also can't be included as a <CFMAILPARAM>. ColdFusion uses the FROM field of the <CFMAIL> tag as the Return-path. You can trick ColdFusion into using a separate FROM field by including a <CFMAILPARAM> tag in the code that specifies another FROM field. The FROM attribute of the <CFMAIL> tag becomes the Return-path parameter, and the <CFMAILPARAM> tag supplies the new FROM field.

We'll follow the exact same steps for this Server Behavior, but we'll include a second code block, which demonstrates one of the nice features of the SBB -- multiple code blocks.

1. Open the Server Behaviors palette, click the plus sign (+) and click New Server Behavior.
2. Give the new Server Behavior a name--"Multipart CFMAIL Email"-- and click OK.
3. Click the plus sign (+) next to the title Code Blocks to Insert. This will bring up a dialog box to allow you to name the code block. You can click OK to keep the default name that is given to the block by the Server Behavior Builder.
4. Type in or copy and paste the <CFMAIL> tag code block into the text area on the SBB, with all parameters as shown in Listing 1. Alternatively, you can insert the parameters one by one as I did in the first example.
5. Next you'll have to choose a location where you want to place the code. In this case, choose to Insert Code "Above the <html> Tag" by choosing it from the dropdown list and the Relative Position attribute should be set to "The Beginning of the File".
6. Add a second code block to the SB and insert the <CFOUTPUT> block in the body of the document. You should set the Insert Code attribute of this one to "Relative to Selection", and set the Relative Position attribute to "Replace Selection". Click Next on the SBB interface.
7. Now you have to set up the types of fields that you want to appear in the Server Behavior interface. You can put them in a different order by clicking the up and down arrows. I have them set up as follows:

| | |
|--------------------------|----------------|
| From Field Nice Name | Text Field |
| From Field Email Address | Text Field |
| To Field | Text Field |
| Reply-To Field | Text Field |
| Return-Path Field | Text Field |
| Subject | Text Field |
| Mail Server | Text Field |
| Text Version Email | URL Text Field |
| HTML Version Email | URL Text Field |

8. Click OK and you're done. The Server Behavior will appear on the Server Behaviors palette.

Now when you want to include a <CFMAIL> tag in the page to send out a multipart email, you can do it with a click and simply fill in the parameters in the SB interface (Figure 6). This practically eliminates the chances of typos or misplaced quotes. Also, you could easily improve on this SB by including a <CFQUERY> tag to allow multiple emails.

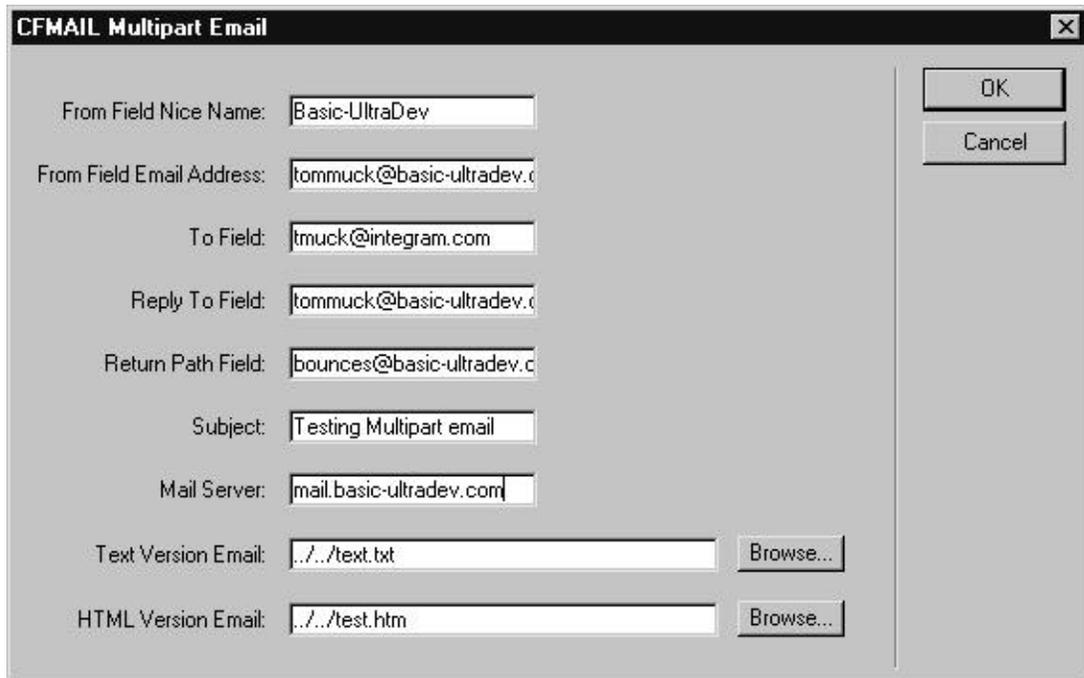


Figure 6: The new CFMAIL Multipart Email server behavior has its own interface now.

The Server Behavior Builder is especially handy for inserting multiple related code blocks into your ColdFusion document. You can insert CF tags, HTML tags, replace attributes of tags, or build complete pages with the SBB. You are only limited by your imagination and desire to increase your workflow. If you haven't tried UltraDev and the Server Behavior Builder, you should download an evaluation copy from Macromedia and give it a try.

Download the completed server behavior at <http://www.basic-ultradev.com/extensions/downloads.asp?id=50>.

(code listing on next page)

Listing 1 (<http://www.basic-ultradev.com/articles/BuildingCFServerBehaviors/listing1.txt>)

```
<CFMAIL TO="@@To Field@"
  FROM="@@Return Path Field@"
  SERVER="@@Mail Server@"
  SUBJECT="@@Subject@">

<CFMAILPARAM NAME="from"
  VALUE='@@From Field Nice Name@@'
@@From Field Email Address@@>'>
<CFMAILPARAM NAME="reply-to"
  VALUE="@@Reply To Field@">
<CFMAILPARAM NAME="mime-version"
  VALUE="1.0">
<CFMAILPARAM NAME="content-type"
  VALUE='multipart/alternative; boundary="-----7C252"'>

-----7C252
Content-Type: text/plain; charset=us-ascii
Content-Transfer-Encoding: 7bit

<CFINCLUDE TEMPLATE="@@Text Version Email@">

-----7C252
Content-Type: text/html; charset=us-ascii
Content-Transfer-Encoding: 7bit

<CFINCLUDE TEMPLATE="@@HTML Version Email@">

-----7C252--

-- End --

</CFMAIL>

<html>
<head>
<title>Test Email</title>
<meta http-equiv="Content-Type"
  content="text/html; charset=iso-8859-1">
</head>
<body bgcolor="#FFFFFF" text="#000000">

<CFOUTPUT>One email sent at
  #DateFormat(now())# #TimeFormat(Now())#
</CFOUTPUT>

</body>
</html>
```

Block 1 for the Server behavior is the complete <CFMAIL> tag

Block 2 for the server behavior is the <CFOUTPUT> tag.

Further Reading:

Dreamweaver UltraDev: The Complete Reference

By Ray West, Tom Muck, Tom Allen

Includes almost 200 pages of detailed descriptions of building server behaviors and other extensions in UltraDev.

<http://www.amazon.com/exec/obidos/ASIN/0072130172/basicultradev-20>

Building Dreamweaver 4 and Dreamweaver UltraDev 4 Extensions

By Tom Muck and Ray West

Step by step guide to building extensions in Dreamweaver and UltraDev.

<http://www.amazon.com/exec/obidos/ASIN/0072191562/basicultradev-20>