

MIME Types, Coldfusion and You

Posted At : August 2, 2006 11:06 AM | Posted By : Mark Kruger

Related Categories: Coldfusion MX 7, Coldfusion Tips and Techniques

Sometimes you need to be able to specify the MIME type using Cfcontent or inline headers to force the file to open in the proper application. If you have a document storage application or a data export application you know what I mean. For example, an engineering firm might upload CAD files, Office files, PDFs, images, and proprietary formats that you have never seen before. If you are like me, you end up writing code that switches against the file extension to set the mime type. Yesterday I saw a note from Dan Switzer from Pengoworks (of [qForms](#) fame) that can get the mime type using the file path. It's not perfect as I'll explain - but here it is:

```
<cfscript>
// get a file with path
myFile      = "D:\test\test.zip";
// find the mime type (string)
myMime      = getPageContext().getServletContext().getMimeType(myFile);
// set the mimetype without using Cfcontent
getPageContext().getResponse().getResponse().setContentType(myMime);
</cfscript>
```

Still not "Automatic"

While this is nifty and works well, it is not a great deal better than a switch statement - except it allows for fewer lines of code. Why? For one thing, it is doing the same thing as your switch statement - it is checking the file extension and finding the associated MIME mapping on the server. That's it. It does not check to see if the file exists. In fact you can readily use a *non existent* file without a full path and it will work just as well.

```
<cfscript>
// get a file with path
myFile      = "someNoneExistentFile.txt";
// find the mime type (string)
myMime      = getPageContext().getServletContext().getMimeType(myFile);
</cfscript>
```

The other thing to remember is that the MIME type comes from the server MIME settings - and not from the file itself. A file doesn't have such an attribute. That means that unless the MIME mapping exists on the server (in the registry under HKEY_CLASSES_ROOT) the function will not return anything useful - or anything at all as we shall see. Since a server doesn't have a bazillion pieces of software installed (hopefully) it doesn't necessarily have hundreds of mime mappings. For example, there is no mime mapping for ".cfm" on my CF server. This can fool you if you do development running a desktop running CF - because your local machine has all the MIME mappings of the crowded subway that is a developers workstation.

Finally, if there is NO MIME mapping the code above does not throw an error. Instead, the variable is simply not set at all. If you try to use it you will get a "variable is undefined" error. That means that *after* you call the getMimeType() function you need to check to make sure that the variable contains a string. The other part of the code is useful for setting the page content type without the need to use Cfcontent.

Streaming Files

This topic came about from the desire to stream files to the browser. 2 other excellent posts were mentioned in the thread. One is by Christian Cantrell on [Using CF to Write Out Binary Data](#) and the other is an awesome snippet of code from Steve Savage on [Streaming Flash FLV Files Using CF](#). This is some very innovative stuff.