

ColdFusion MX 7, Learning the new features

What's new in ColdFusion MX 7

Finally a real "developer" version is released of ColdFusion. The release of MX a few years ago was more a performance update by making it a Java based server instead of the "full of memory leaks" C++ version. With the capabilities of Java, the ColdFusion team managed to create some very helpful features, such as reporting capabilities, FlashPaper 2/PDF creation, Rich Forms, new Charting engine, improved search engine (Verity), Dreamweaver MX 2004 server management integration, Gateways, Administrator API and lots and lots of other features.

So let's start with one of the, in my eyes, best productivity enhancements, Rich Forms.

Rich Forms

With the ColdFusion Rich Forms you can easily create complex forms, generated in plain HTML, Xforms (XML) or Flash forms. Let's take a look at an example of a Rich Form in Flash format.

```
<cfform name="foo" action="process.cfm" method="post" format="flash" skin="haloBlue"
width="390" timeout="10" preloader="true">
  <cfformgroup type="tabnavigator">
    <cfformgroup type="page" label="Message Inbox">

    </cfformgroup>
    <cfformgroup type="page" label="Account settings" height="200">

    </cfformgroup>
    <cfformgroup type="page" label="Other options">

    </cfformgroup>
  </cfformgroup>
</cfform>
```

This piece of code will generate a flash form with a blue Halo themed skin, and will have a preloader when the form is requested. The form will contain a tabbed navigator with 3 tabs, namely, "Message Inbox", "Account settings" and "Other options". You can also create this sort of form with XML forms. Skin you XML form with one of the XSL files shipped with CFMX 7. Or make a copy of one them ("[ColdFusion root]/CFIDE/scripts/xsl/"), copy and rename a file .xsl file ("e.g. myStyle.xsl"), modify the colors and etc and fill in the filename of your sheet in the **SKIN** parameter. Viola your own look and feel XML form. This is something you need to play with to really learn what it can do.

Below is a screenshot of the completed flash form that I started with in the code above.

Flash Forms and ActionScript 2.0

When building a Flash form, you can use ActionScript 2.0, in a sort of Flex way. Only in ColdFusion there are a few restricted ActionScript keywords. Be sure to check the CFMX 7 documents for more details about using ActionScript 2.0 in the ColdFusion Rich Forms.

Message Inbox Account settings Other options

	Subject	date	time	
1	Foo message 1	Monday 14 Feb 2005	22:10	▲
2	Foo message 2	Sunday 13 Feb 2005	21:10	☰
3	Foo message 3	Saturday 12 Feb 2005	20:10	
4	Foo message 4	Friday 11 Feb 2005	19:10	
5	Foo message 5	Thursday 10 Feb 2005	18:10	
6	Foo message 6	Wednesday 09 Feb 2005	17:10	
7	Foo message 7	Tuesday 08 Feb 2005	16:10	▼

Subject: Foo message 4

Date: Friday 11 Feb 2005 19:10

Message 4

Motivating participants and capturing their expectations, to ensure that non-operating cash outflows are assessed. Working through a top-down, bottom-up approach, whenever single-loop learning strategies go wrong, the balanced scorecard, like the executive dashboard, is an essential tool. Big is no longer impregnable the components and priorities for the change program exploiting the productive lifecycle.

PDF and FlashPaper 2 documents

One of the other big features is the ability to create PDF/FlashPaper 2 documents. It's offered in such an easy way, that you will really like and use this function. Let me give some example code how this is done.

```
<cfdocument format="flashpaper">
    That's all!
</cfdocument>
```

That's all there is to generate the most simple type of document, in this case a FlashPaper document. If you want it to be a PDF, just change the format parameter to "PDF" and viola a PDF it is! So this was pretty easy to do, right? Let me show you another simple example with some more code. The following code will generate a document with a header and footer. The cfdocument brings also a new scope to ColdFusion, namely "cfdocument" (sigh, ColdFusion is just so easy!). With this scope you can display the page number, and/or display the number of total pages the document has. This is also demoed in the code bellow. Check the footer and you'll see "#cfdocument.currentpagenumber# of #cfdocument.totalpagecount#". It's almost the same as with the cfquery scope (queryname.CurrentRow and queryname.RecordCount).

```
<cfdocument format="flashpaper" orientation="landscape">
  <cfdocumentitem type="header">
    <center>
      <div style="width:100%; background-color:#000000; color:#66CCFF; ">
        Hello! I'm a header...
      </div>
    </center>
  </cfdocumentitem>
  This is page 1
<cfdocumentitem type="pagebreak" />
  And this is page 2... Duh!
  <cfdocumentitem type="footer">
    <center>
      <div style="background-color:#66CCFF; width:100%; ">
        <cfoutput>
page #cfdocument.currentpagenumber# of #cfdocument.totalpagecount# <br />
by Michael van Leest
        </cfoutput>
      </div>
    </center>
  </cfdocumentitem>
</cfdocument>
```

And I hear you think, "Why did he put in, **orientation="landscape"??**". This will turn the page 90 degrees, so it will have the same "landscape" style as for instance word or any other document editor that can display/print a page. Pretty easy to do. There are a lot of other parameters and some PDF specific parameters for creating password protected content and etc. There is only 1 thing I want to show you and that is how you can **SAVE** your document to your disk for any type of distribution, history management or whatever you want to do with it. Here's the example how you can save the document in the same directory as the page is executed.

```
<cfdocument format="flashpaper" filename="testFP.swf">
  Hello world!
</cfdocument>
```

If you want to save somewhere else on your drive, it will look something like this.

```
<cfdocument format="flashpaper" filename="C:\cfdocuments\testFP.swf">
  Hello world!
</cfdocument>
```

So that's really easy and to switch from PDF to Flashpaper is just 1 parameter and visa versa.

cfdocument supported HTML

The cfdocument supports HTML 4.01, XML 1.0, DOM Level 1 and 2, CSS 1 and 2.

cfdocument performance

Opening a PDF document is far more slower for the end user then opening a FlashPaper 2 document. Although PDF is a well known standard in cross platform documents, the flash player has a much higher penetration figure than the PDF reader. It also has the benefit that the document size is much smaller, thus loads quicker. Be sure to inform the end user about what flashpaper is and how they can use it.

cfdocument differences in ColdFusion MX 7 versions

There is a performance limitation with the cfdocument tag in the "CFMX 7 Standard edition". It processes the cfdocument requests one at a time. The Enterprise edition can publish multiple documents at the same time.

cfdocument extensions

Don't forget to add the extension of the document when saving it somewhere on your disk. For FlashPaper append the extension ".swf" and for PDF append the extension ".pdf".

ColdFusion Reporting

This is divided in 2 separate sections, namely the "ColdFusion Report Builder" that creates .cfr files, and the output part that generates the reports in different formats and optional adds report parameters. Some of you may know "Crystal Reports" and ColdFusion Reports look just like them, only better. One of the benefits of using ColdFusion Reports is that you can use ColdFusion variables in your report such as session and application variables. With for instance session variables you can create user specific reports, for example sales reports for the specific user (by checking his user id that is stored in the session). Also a dynamic query is one of the plus points. You can build an interface where the user can put in minimums and maximums of different data, such as dates, months, prices or whatever the query gets from the database. In this way you're much more flexible with just 1 report template. If you wanted to do this with "Crystal Reports", you had to make a lot of different reports to get the same kind of diversity and flexibility in reports. Because reporting is such a big and new topic, I'll just tell you about the basic steps you need to walk through.

The Report Builder Part

The report builder itself starts with a wizard selection, which is really helpful especially the first time you'll use the ColdFusion Report Builder. One of the 2 wizards will help you setup a connection to the ColdFusion server. After that you can run the other wizard and walkthrough the following steps:

1. Select a measurement unit (i.e. cm's, pixels etc)
2. select a datasource (DSN) from which you query the data. After selecting a DSN, you can build your query which is much easier to create pretty complex stuff without a lot of hassle then the Dreamweaver MX 2004 version of it.
3. After creating and saving the query and selecting the "printed fields" and the "non-printed fields", you can select how to group your report.
4. Specify which layout you want to use.
5. Define the "styling" of your document
6. Choose the "Report Theme".
7. Give the report a name and optional a header and footer.
8. Press FINISH

Ok it looks pretty strange if you've never seen a report before, but if you look good and dig into all the setting of the document and play around with columns, titles and etc you'll find a way to display everything how you want it and where you want it.

The output part

So far for the Report Builder part. Lets move on to the displaying part of your report. Let me show you how to display your report with some example code.

```
<cfreport template="report-1.cfr" format="flashpaper">
</cfreport>
```

The "Template" parameter needs to have the full path from the webroot to the .cfr file, for example "/reports/report1.cfr". The parameter "format" is also required and has 3 options to choose from. PDF, Flashpaper AND EXCEL documents! Yes you can even create Excel documents. And again you can save them somewhere just in the same way I showed with the cfdocument tag.

cfreport performance

Opening a PDF document is far more slower for the end user then opening a FlashPaper 2 document. Although PDF is a well known standard in cross platform documents, the flash player has a much higher penetration figure than the PDF reader. It also has the benefit that the document size is much smaller, thus loads quicker. Be sure to inform the end user about what flashpaper is and how they can use it.

cfreport differences in ColdFusion MX 7 versions

There is a performance limitation with the cfreport tag in the "CFMX 7 Standard edition". It processes the cfreport requests one at a time. The Enterprise edition can publish multiple reports at the same time.

cfreport Builder RDS Login

There is an "username" field for specifying the RDS login. Leave this field blank in order to let the Report Builder function properly.

Application.cfc

Some of you might think this is a more "advanced" topic, but to be honest, it isn't. It's really easy to easy and works better than the normal "Application.cfm". The Application.cfc is created so developers can build more Object Orientated applications. This is a good start to learn a more OO way of programming without any difficult, not understandable instructions. If you know how to use and build a .cfc (ColdFusion Component), you know 90% of how to build a Application.cfc.

To setup a Application.cfm, you normally use the "cfapplication" tag. Now we'll set those arguments at the top of the Application.cfc as "This" variables. Here's a small example:

```
<cfcomponent>
    <cfset This.name="TestApplication" />
    <cfset This.Sessionmanagement="True" />
    <cfset This.SessionTimeout="#CreateTimeSpan(0,0,10,0)#" />
    <cfset This.ApplicationTimeout="#CreateTimeSpan(0,0,30,0)#" />
    ...

```

It's written in a different way as normally in a CFMX 6 application, but it makes sense if you know how to use a "cfapplication" tag and know how to create a CFC. To make the developers life easier and the code much more cleaner, there are 8 different methods, namely:

- onApplicationEnd (executes whenever the Application will time out)
- onApplicationStart (executes whenever the Application starts for the first time)
- onError (executes whenever an error occurs that isn't caught by a cftry/cfcatch block)
- onRequest (executes when the onRequestStart ends, this can filter any requested content)
- onRequestEnd (executes whenever a request is processed)
- onRequestStart (executes when a request is done, after onApplicationStart and onSessionStart ends)
- onSessionEnd (executes when the session times out)
- onSessionStart (executes when a session is started)

Anyone who uses the Application.cfm and CFC's, will love this way of application management. This topic is also big to dig into, but the get started with it, I recommend the ColdFusion MX 7 reference that's

included in the Dreamweaver MXP. Also play a bit around with the methods, to really know whenever something is executed or not and if it's executed, in which order. There are a lot of notes that you'll need to keep in mind, so please check the reference guide first, before start swearing and throwing with your laptop. Because I did almost something like that and it wasn't necessary if I just had read the manual ;-).

A minor (but really really, really useful) function

One of "the smallest" enhancements, but one of my favourites, is the "IsValid()" function. This function checks if a value is valid. For example to check if something is a valid UUID you can do this.

```
<cfif IsDefined("URL.ProductID") AND IsValid("UUID", URL.ProductID)>
    <!--- Execute some code --->
</cfif>
<cfelse>
    <!--- Redirect to a error message --->
</cfif>
```

Normally this needed a much bigger and longer validation to check if the provided data was really a UUID. But now just check if the variable exists (Good coding practice) and check if it's the good type of data. You can check if the variable is a valid array, binary, Boolean, credit card, date or time, email and etc. etc. If you want to check a range, you *can* check a range. Look at the following example.

```
<cfif IsDefined("URL.showDetail") AND IsValid("range", URL.showDetail, 1,
qGetRecords.RecordCount)>
    <!--- Redirect and show the requested record details from the query --->
</cfif>
<cfelse>
    <!--- Show "sorry, incorrect detail ID given" --->
</cfif>
```

Even if you know how to use "regex", you can check if it's valid to the regex pattern. But that's a bit too deep into this tiny topic if you ask me.

That's all for now!

So yeah, that's all for now. I even haven't started about ColdFusion Gateways, Improved Verity search engine, The administrator API, New charting engine, Debugging enhancements, Strong encryption, Enhanced webservice support, Dreamweaver MX 2004 extension, Enterprise manager, Sourceless deployment and more of that kind of stuff. Man this version is packed with new capabilities and I'll be writing about specific topics in the very near future. So hopefully you liked my *FIRST* article and I would love to get some feedback about it. Just drop a message in the ColdFusion forum of DMXzone or drop me an email.

If you have any questions about CFMX 7, take a look at these websites:

- DMXzone ColdFusion Forum (http://www.dmxzone.com/forum/forum.asp?FORUM_ID=9&CAT_ID=5&Forum_Title=CFML)
- My weblog (<http://www.wantonline.com/blog/?refer=DMXtut1>)
- Macromedia LiveDocs (<http://livedocs.macromedia.com/ColdFusion/7/>)
- Macromedia ColdFusion Forum (<http://www.macromedia.com/cfusion/webforums/forum/index.cfm?forumid=1>)
- Or just send me an email (mvanleest@gmail.com)

Thanks for reading!