Wrapping a Fusebox App in Farcry

Posted At : September 27, 2006 3:25 PM | Posted By : Mark Kruger
Related Categories: farcry

We have an internal project tracking application based on Tracking Tools by Phil Cruz. We have enhanced it by adding hours tracking, milestones, reporting and extra notification features. We use it to communicate internally *and* to communicate with our customers regarding work that is underway. It has worked well up to this point.

Recently I became heavily involved in several Farcry projects - as you know if you read my blog on The Greatest Coldfusion Product Ever. I hit upon the idea of wrapping Farcry around my tracking-tools application (written in Fusebox). I could get the advantages of content management and post news and updates and other information to my staff and to my customers, and still be able to leverage the tracking tools application that we have modified. It seemed like a good idea - so I got started on it right away. Here's how I did it and what I learned.

Here are the steps in order:

- **Step 1: copy the whole app to the includedObj folder** - This could actually be anywhere you like - as long as you can include it in a display template. Make sure and work your way through the application settings and adjust an paths or roots you might see there.
- **Step 2: Create a new DisplayPage template** - This page should be a copy of whatever skin you are using. It will get some special attention.
- **Step 3: Create an dmHTML page in Farcry and assign it to that template.**
- **Step 4: At the top of the template include the index page of the FB ap.**
- **Step 5: Open the FB index.cfm page and add a cfinclude for the FB's application.cfm**
- **Step 6: Open the FB's application.cfm and remove the "cfapplication" tag** - the application will now run in the context of your farcry application.
- **Step 7: Edit the "layout" template of the FB application** - this could be a little different depending on the implementation of FB - but somewhere there is probably a variable like "layout" that contains the main content of the page. You will want to copy this variable to the variables scope (name it something else like "fbcontent") because it is sometimes altered by later processes. Then, in your template add the outputted variable underneath the "#stobj.body#" output.
- **Step 8: Add the ojbectID to all URLs** - There is usually a variable that represents the main url used by the application. In my tracking tools applicatin it is called "selfwithtoken" and it is set early in the page processing. I merely had to find that set statement and add "&objectid=xxxxxxxxx" to it (using the ojbjectid from the page I created in step 3). From that point on all the FB requests will run through my content managed page. You could just as easily use a URL variable for objectid to use different pages. Just make sure and use the configured FB template.
- **Step 9:** - test and debug. It is rare to find a strictly implemented fusebox application. It is likely you will need run it through it's paces and fix minor issues throughout the application.

One problem I'm currently trying to solve is session management. Since the FB ap copies everything to the request scope it makes it difficult to see where items are coming from. I'm working through it and I'll try and post an update.

Tip: Dynamic containers

One nice thing about doing it this way is that you can actually implement containers based on your fuse actions. In my application I have a helper sidebar that I want to contain different information based on the fuseaction. I used code like this:

```
<cfparam name="url.cmd" default="start"/>
    <cfif isDefined('url.fuseaction')>
        <cfset url.cmd = url.fuseaction>
    <cfelseif isDefined('form.fuseaction')>
        <cfset url.cmd = form.fuseaction>
    </cfif>
    <con:container label="help_#replace(url.cmd,'.','-')#">
```

The result is that the container has a different label based on all the available fuse actions. I can simply go to edit mode and write appropriate context sensitive help using a text rule - or a news rule and categories or whatever. I'm sure some of the Farcry gurus out there can add some comments and tell me how much easier I could have made it by doing A or B. Fire away - I can take it.