

May 26, 2008

## ColdSpring ColdboxFactory: DSN and Mail settings

My second day into figuring out Coldbox, here's a little something cool. Putting it simply, using ColdSpring, in the CS.xml you can define a 'coldboxFactory' bean to use in creating other beans that define this factory's methods as beans also, then pass that bean into a class bean's property as a bean reference.

Actually, that doesn't sound simple as it is, so here's an example. Remember that I'm a newbie, so be nice.

coldbox.xml.cfm

```
<MailServerSettings>
  <MailServer>mail.urserver.com</MailServer>
  <MailPort>25</MailPort>
  <MailUsername>username</MailUsername>
  <MailPassword>password</MailPassword>
</MailServerSettings>

<Datasources>
  <DataSource alias="DSNAlias" name="dsnName" dbtype="mysql" username="name" password="pass" />
</Datasources>
```

I've set mail settings and data source settings here in the main Coldbox config. I'll be using Coldspring for IoC and so I'll need to be able to pass this around in my beans.

Coldspring.xml

```
<beans>
  <bean id="coldboxFactory" class="coldbox.system.extras.ColdboxFactory" />
  <bean id="ConfigBean" factory-bean="ColdboxFactory" factory-method="getConfigBean" />

  <bean id="dsnBean" factory-bean="ColdboxFactory" factory-method="getDataSource">
    <constructor-arg name="alias">
      <value>DSNAlias</value>
    </constructor-arg>
  </bean>

  <bean id="cfctest" class="handlers.cfctest">
    <property name="dsnBean">
      <ref bean="dsnBean" />
    </property>
    <property name="ConfigBean">
      <ref bean="ConfigBean" />
    </property>
  </bean>
</beans>
```

This coldboxFactory has several methods that interface with the coldbox config file, among other things I'm sure. Check out the [API](#) for more info. But basically I create the coldboxFactory bean, then create other beans of the factory's methods. Then notice how I pass those beans into the handler beans by reference. These 'properties', as it turns out, are setters in the cfc, such as setDsnBean and setConfigBean. Then I just put them in variables for my other methods.

cfctest.cfc

```
<cfcomponent output="false">
  <cffunction name="setDsnBean" access="public" returnType="void">
    <cfargument name="DsnBean" type="Any" required="true"/>
    <cfset variables.dsn = arguments.DsnBean.getName()>
  </cffunction>

  <cffunction name="setConfigBean" access="public" returnType="void">
    <cfargument name="configBean" type="Any" required="true"/>
    <cfset variables.mailName = arguments.configBean.getKey("mailUsername")>
  </cffunction>

  <cffunction name="capitalize" access="public" returnType="string">
    <cfargument name="string" type="string" required="true"/>
    <cfreturn Ucase(variables.mailName & " " & arguments.string)/>
  </cffunction>
</cfcomponent>
```

Don't dis my cfc, I'm just playing around and testing stuff. Notice the setters that I mentioned before. If you do a dump/abort on these at run time you'll see what use they are.

general.cfc

```
<cffunction name="index" access="public" returnType="void" output="false">
  <cfargument name="Event" type="coldbox.system.beans.requestContext">

  <!-- Logic -->
  <cfset string = "Welcome!">
  <cfset string = variables.ioc.getBean("cfctest").capitalize(string)>
  <cfset Event.setValue("welcomeMessage",string)>

  <!-- Display -->
  <cfset Event.setView("home")>
</cffunction>
```

And finally here's the handler that uses the IoC plugin I set in variables in the handler's init function that gets the bean set in Coldspring and runs the method that utilizes all the stuff we just went through. Check out [Coldbox's ColdSpring documentation](#) for more information.

There are other factories to use I'm guessing and other factory methods within those. You can use all the methods in the coldbox.system.controller as beans also. But like I said, I'm just discovering this stuff right now. Frankly I need to think over whether or not passing bean reference properties to all my CS beans is the most effective way of using global configs any ways. Seems rather repetitive as opposed to putting such things in a request

scope through onRequestStart or something.

Posted At : 6:06 AM. | Posted By : clint317