

# Dynamic Datasources with Transfer and Coldspring

Posted At : July 12, 2007 10:44 AM | Posted By : Bob Silverberg

Related Categories: ColdFusion, Coldspring, Transfer

I have different datasource names on my local machine, in my dev environment and in my production environment, and I wanted to figure out a way to automatically tell transfer which datasource to use based on the server it's running on. I couldn't find any programmatic way of doing this with Transfer, as it seems to require a datasource.xml file. I decided to blog my solution, as it may be useful to others, or perhaps someone will point out a much better way to do it.

Here's what I'm doing:

I create 3 separate xml files, datasource\_localhost.xml, datasource\_dev.xml and datasource\_prod.xml. Each one contains the name of the appropriate datasource.

In my Coldspring.xml file, where I define the Transfer bean, I use this code:

```
<bean id="Transfer" class="transfer.TransferFactory">
  <constructor-arg name="datasourcePath"><value>$(TransferDataSourcePath)</value></constructor-arg>
  <constructor-arg name="configPath"><value>/_MyModel/config/transfer.xml.cfm</value></constructor-arg>
  <constructor-arg name="definitionPath"><value>/_MyModel/config/transfer</value></constructor-arg>
</bean>
```

In my CF code which creates the TransferFactory, I do the following:

```
<cfset TransferDataSource = "datasource_prod.xml.cfm" />
<cfif getPageContext().getRequest().getServerName() EQ "localhost">
  <cfset TransferDataSource = "datasource_localhost.xml.cfm" />
<cfelseif getPageContext().getRequest().getServerName() CONTAINS "dev.mydomainname">
  <cfset TransferDataSourcePath = "datasource_dev.xml.cfm" />
</cfif>

<cfset defaultProperties = StructNew() />
<cfset defaultProperties.TransferDataSourcePath = "_MyModel/config/" & TransferDataSource />
<cfset serviceFactory = CreateObject("component","coldspring.beans.DefaultXmlBeanFactory").init (defaultProperties=defaultProperties) />
```

I'm setting other default properties, unrelated to Transfer, based on ServerName, but I left them out in the interest of brevity.

It does seem a bit ugly, but it works very well. I realize that ANT is probably the answer to this, but, only working two days per week, I haven't found the time to figure ANT out yet, so I came up with this solution in the meantime.