

Thread: SES routes.cfm on Apache	
Created on: 01/21/08 12:21 PM <span style="float: right;">Replies: 6</span>	
<p><b>pwpeterson</b> Rookie</p> <p><b>Joined:</b> 06/11/07 <b>Posts:</b> 13</p>	<p><b>SES routes.cfm on Apache</b> 01/21/08 12:21 PM</p> <p>I hope somewhat can help me... I moved some CB apps to an apache/tomcat server and I am getting this message when I invoke them:</p> <p>Error including config file: Could not find the included template /cfusion/[APPNAME]/config/routes.cfm.</p> <p>This happens with the dashboard as well.</p> <p>Any ideas why CF or CB can't find the routes.cfm file. It IS there...</p> <p>pwp</p> <p style="text-align: right;"><a href="#">Link</a>   <a href="#">Top</a>   <a href="#">Bottom</a></p>
<p><b>Imajano</b> Veteran Master</p> <p><b>Joined:</b> 01/29/05 <b>Posts:</b> 1209</p>	<p><b>RE: SES routes.cfm on Apache</b> 01/21/08 1:09 PM</p> <p>Try using an absolute path or just reinitializing the application.</p> <p>I run my stuff on apache, but never have this problem. Its hard without seeing any of the code or your setup.</p> <p>Luis Majaño -----</p> <p style="text-align: right;"><a href="#">Link</a>   <a href="#">Top</a>   <a href="#">Bottom</a></p>
<p><b>pwpeterson</b> Rookie</p> <p><b>Joined:</b> 06/11/07 <b>Posts:</b> 13</p>	<p><b>RE: SES routes.cfm on Apache</b> 01/21/08 4:38 PM</p> <p>I tried installing CF* last week and ran into some problem so I reinstalled CF7.</p> <p>My Tomcat's appbase is C:\webapps. I just added a CF mapping of /cfusion =&gt; c:\webapps\cfusion and it started working again.</p> <p>(Interesting my other CF apps worked fine before the mapping.)</p> <p>I still can't start the cfadmin from CB's dashboard.</p> <p>pwp</p> <p style="text-align: right;"><a href="#">Link</a>   <a href="#">Top</a>   <a href="#">Bottom</a></p>
<p><b>Hayesdata</b></p> <p><b>Joined:</b> 06/11/09 <b>Posts:</b> 4</p>	<p><b>RE: SES routes.cfm on Apache</b> 06/11/09 5:32 PM</p> <p>I'm currently running into the same issue whilst working in Railo on Tomcat. I've tried every possible combination of mappings and xml config changes, however the issue seems to be with Tomcat.</p> <p>Example URL:</p> <p><code>http://localhost:8080/shoutEngine/index.cfm/users/login</code></p> <p>What appears to be happening is that Tomcat is evaluating the entire path and not stopping at the index.cfm as it should.</p> <p>If you have any ideas please let me know.</p> <p style="text-align: right;"><a href="#">Link</a>   <a href="#">Top</a>   <a href="#">Bottom</a></p>
<p><b>Imajano</b> Veteran Master</p> <p><b>Joined:</b> 01/29/05 <b>Posts:</b> 1209</p>	<p><b>RE: SES routes.cfm on Apache</b> 06/11/09 5:43 PM</p> <p>Tomcat implements the servlet spec to the T. Therefore, url's like this:<code>http://localhost:8080/shoutEngine/index.cfm/users/login</code> are NOT allowed by the servlet specification.</p> <p>Therefore, tomcat does not support them. You will have to run it through a rewrite filter or engine like <code>mod_rewrite</code> or <code>j2ee</code> rewrite filter to make them work like that.</p> <p>I would suggest looking at <code>mod_rewrite</code> or <a href="https://simpleurlrewrite.dev.java.net/">https://simpleurlrewrite.dev.java.net/</a> or <a href="http://tuckey.org/urlrewrite/">http://tuckey.org/urlrewrite/</a></p> <p>Luis Majaño -----</p> <p style="text-align: right;"><a href="#">Link</a>   <a href="#">Top</a>   <a href="#">Bottom</a></p>
<p><b>analogdigital</b> Veteran Master</p> <p><b>Joined:</b> 04/26/09 <b>Posts:</b> 121</p>	<p><b>RE: SES routes.cfm on Apache</b> 07/01/09 3:05 AM</p> <p>I have accomplished the following in ColdBox (notice no index.cfm):</p> <p><code>http://foo.com/myHandler/action</code> <code>http://foo.com/123</code> <code>http://foo.com/user/25</code> <code>http://foo.com/user/delete/28</code> etc...</p> <p>Using three different methods. Tuckey UrlRewrite with Tomcat, <code>mod_rewrite</code> with Apache to Tomcat, as well as <code>mod_jk</code> from Apache to Tomcat. All three work identically when configured correctly. I have verified with extreme testing.</p> <p>If you use <code>mod_jk</code>, the Tuckey UrlRewrite filter loaded in Tomcat as a servlet will do the rewrite without any rewrite rules needed in Apache:</p> <pre>&lt;rule&gt; &lt;from&gt; ^/index.cfm/(.*)\$&lt;/from&gt;</pre> <p style="text-align: right;"><a href="#">Link</a>   <a href="#">Top</a>   <a href="#">Bottom</a></p>

```

<to last="true">/index.cfm?PATH_INFO=$1</to>
</rule>

<rule>
<from>^/index.cfm</from>
<to last="true">/index.cfm</to>
</rule>

<rule>
<from>^/\?(.*)$</from>
<to last="true">/index.cfm?$1</to>
<set name="urlrewrite.originalRequestUri">{%{request-uri}</set>
<set name="urlrewrite.originalQueryString">{%{query-string}</set>
</rule>

<rule>
<from>^/(.*)$/?(.*)$</from>
<to last="true">/index.cfm?PATH_INFO=$1&{%{query-string}</to>
<set name="urlrewrite.originalRequestUri">{%{request-uri}</set>
<set name="urlrewrite.originalQueryString">{%{query-string}</set>
</rule>

<rule>
<from>^/(.*)$</from>
<to last="true">/index.cfm?PATH_INFO=$1&{%{query-string}</to>
<set name="urlrewrite.originalRequestUri">{%{request-uri}</set>
<set name="urlrewrite.originalQueryString">{%{query-string}</set>
</rule>

<rule>
<from>/(.*)$</from>
<to last="true">/index.cfm?PATH_INFO=$1</to>
<set name="urlrewrite.originalRequestUri">{%{request-uri}</set>
<set name="urlrewrite.originalQueryString">{%{query-string}</set>
</rule>

```

(I also have flex2gateway, etc, but it was too much to put here on the post)

If you use Tomcat standalone, the same rules for Tuckey UrlRewrite will work. The only difference is that Apache forwards its requests with jkMount but the rules are the same as far as Tomcat is concerned, it doesn't need Apache to have sent it the URL to rewrite it. If you access the Tomcat server directly with a URL, these rewrites will happen the same as if you came from Apache.

If you let Apache do the basic rewriting with mod\_proxy, you can ProxyPass to Tomcat over AJP which will then allow UrlRewrite to do the more advanced rules (above)...

```

ProxyPreserveHost On
ProxyPassReverse / ajp://localhost:8009/

RewriteEngine On
RewriteRule ^/(.*\.cf[cm])$ ajp://localhost:8009/$1 [P]
RewriteRule ^/(.*)$ ajp://localhost:8009/$1 [P,L,R]

```

You can also do all of the rewriting in Apache and skip UrlFilter if you do this:

```

RewriteRule ^/(.*\.cf[cm])$ ajp://foo.com:8009/$1 [P]

RewriteRule ^/(.*\.cf[cm])/(.*)$ ajp://foo.com:8009/$1?path_info=$2&{%{QUERY_STRING}} [P]

RewriteRule ^/index.cfm ajp://foo.com:8009/index.cfm [P]

RewriteRule ^/(.*)$ ajp://foo.com:8009/$1?path_info=$1&{%{QUERY_STRING}} [P]

```

Instead of using the default SES interceptor, I simply created my own that extended the original and overwrote the preProcess() method. The reason I did this was because when you do this kind of rewriting, you lose CGI.PATH\_INFO (which ColdBox uses for SES routes), so I passed it along as a URL variable. If you are using CF8 on IIS or not doing any rewriting, then you can just intercept with the normal SES interceptor.

I added this at the top of the method:

```

<cfif #isDefined("url.path_info")#>
<cfset MYPATH_INFO = "#url.path_info#">
<cfelse>
<cfset MYPATH_INFO = "">
</cfif>

```

Then replaced the normal cleanedPathInfo with this:

```

var cleanedPathInfo = #MYPATH_INFO#;
var cleanedScriptName = trim(reReplacenocase(getCGIElement('script_name'), "[\\]index\.cfm", ""));

```

This allows routes to use the PATH\_INFO I passed over in the URL since it can't find the right value from the CGI scope.

This allows "perfect" URLs, since you can map routes to things like:

```
http://www.foo.com/username to:
```

```

pattern=":userID",
handler="users",
action="getUserProfile"

```

	<a href="#">Link</a>   <a href="#">Top</a>   <a href="#">Bottom</a>
<b>Hayesdata</b>	<b>RE: SES routes.cfm on Apache</b> 07/01/09 6:39 PM
<b>Joined:</b> 06/11/09 <b>Posts:</b> 4	Thanks analogdigital I'll try those filters out this evening.
	<a href="#">Link</a>   <a href="#">Top</a>   <a href="#">Bottom</a>