**Hi Team,**

**FCAC\_OMNIA**

**Step 1:**

**Copy and extract tomcat stored in following location.**

[**\\10.8.18.15\Storage\Tony\tomcat**](file:///\\10.8.18.15\Storage\Tony\tomcat)

**Step 2:**

**Copy following libraries into \FCACMobileOmnia\src\main\webapp\WEB-INF\lib**

**Library dependencies:**

* Use latest bfrc-framework.jar
* Use jboss-all-client.jar – This jar has class dependencies of J2EE libraries ([\\10.8.18.15\Storage\Tony\tomcat](file:///\\10.8.18.15\Storage\Tony\tomcat))

**Step 3:**

**JDBC Data Source Configuration in Tomcat. Note: It is already configured in the above specified tomcat zip file.**

**Changes in \conf\context.xml**

<ResourceLink name="jdbc/CrmData"

global="jdbc/CrmData"

type="javax.sql.DataSource" />

**Changes in \conf\server.xml**

<GlobalNamingResources>

………

**<Resource name="jdbc/CrmData" auth="Container"**

**type="javax.sql.DataSource" driverClassName="oracle.jdbc.OracleDriver"**

**url="jdbc:oracle:thin:@10.8.18.15:1521:bsro"**

**username="rtms\_webdb" password="rtms\_webdb" maxActive="20" maxIdle="10"**

**maxWait="-1" />**

  </GlobalNamingResources>

**Changes in FCAC Omnia application, JDBC Datasource entry on DAO configuration**

\FCACMobileOmnia\WEB-INF\classes\firestone\spring\daoContext.xml

\FCACMobileOmnia\WEB-INF\spring\daoContext.xml

<bean id="bfrcDataSource" class="org.springframework.jndi.JndiObjectFactoryBean">

                                <property name="jndiName" value="**java:/comp/env/jdbc/CrmData**" />

                                <property name="resourceRef" value="false" />

</bean>

**Step 4:**

**Change application log4j.xml file in the directory WEB-INF\classes\ to add application specific log on tomcat logs directory**

<param name="file" value="${project.artifactId}.log" />

Change as

<param name="file" value="${catalina.home}/logs/fcacmobile.log" />

**Step 5:**

Running FCAC Omnia application on root context - **Changes in \conf\server.xml**

<Host name="localhost"  appBase="webapps"

            unpackWARs="true" autoDeploy="true">

…………………

**<Context path="" docBase="FCACMobileOmnia">**

**<!-- Default set of monitored resources -->**

**<WatchedResource>WEB-INF/web.xml</WatchedResource>**

**</Context>**

</Host>

1. Remove the out-of-the-box ROOT/ directory from tomcat and rename your war file to ROOT.war before deploying it.
2. Deploy your war as (from your example) war\_name.war and configure the [context root](http://tomcat.apache.org/tomcat-6.0-doc/config/context.html) in conf/server.xml to use your war file :

<Context path="" docBase="war\_name" debug="0" reloadable="true"></Context>

/////////////////////////////////////////////////////////////////

To remove logger in console

.handlers = 1catalina.org.apache.juli.FileHandler, java.util.logging.ConsoleHandler

to

.handlers = 1catalina.org.apache.juli.FileHandler

That stops anything getting written to catalina.out

///////////////////////////////////////////////////////////////

**FCAC\_DESKTOP**

1. For Data base connectivity add the below lines in the file **/FCAC/src/main/webapp/WEB-INF/spring/daoContext.xml**

<bean id="bfrcDataSource" class="org.springframework.jndi.JndiObjectFactoryBean">

<property name="jndiName" value="java:/comp/env/jdbc/CrmData"/>

<property name="resourceRef" value="false" />

</bean>

1. Changes error page path in FCAC web.xml file (/WEB-INF/views/error/404new.jsp)
2. Changes in reading properties file from java class to applicationContext.xml

<bean id="resourceLoader" class="org.springframework.core.io.DefaultResourceLoader" />

<bean id="propertyConfigurer" class="com.bfrc.framework.util.PropertyConfigUtil">

<constructor-arg type="org.springframework.core.io.ResourceLoader" ref="resourceLoader"/>

<constructor-arg type="java.lang.String" value="web-config"/>

<property name="defaultLocation" value="/serverconfig/" />

</bean>

<bean

class="org.springframework.beans.factory.config.PropertyPlaceholderConfigurer">

<property name="location">

<value>/WEB-INF/serverconfig/web-config.properties</value>

</property>

</bean>

1. Comment websphere JTA Transactions in daoContext file

<!-- JTA Transactions -->

<!--<bean id="jtaTransactionManager" class="org.springframework.transaction.jta.WebSphereUowTransactionManager" />

<bean id="jtaTransactionManagerTransactionManager" class="org.springframework.beans.factory.config.MethodInvokingFactoryBean">

<property name="targetObject" ref="jtaTransactionManager"/>

<property name="targetMethod" value="getTransactionManager"/>

</bean>-->

1. Remove the xml folder under the specified folder ( /FCAC/src/main/webapp/WEB-INF/views/completely-firestone/xml ) and save it to the specified location (/FCAC/src/main/resources/xml/) and change the path in property file
2. To change oracle+websphere blob handler to tomcat default blob handler in daoContext file.

<!--<bean id="nativeJdbcExtractor" class="org.springframework.jdbc.support.nativejdbc.WebSphereNativeJdbcExtractor"/>-->

<bean id="nativeJdbcExtractor" class="org.springframework.jdbc.support.nativejdbc.CommonsDbcpNativeJdbcExtractor"/>

1. To remove JTA transaction because Tomcat does not support JTA Transaction in daoContext.xml

<!--<bean id="jtaTransactionManager" class="org.springframework.transaction.jta.WebSphereUowTransactionManager" />

<bean id="jtaTransactionManagerTransactionManager" class="org.springframework.beans.factory.config.MethodInvokingFactoryBean">

<property name="targetObject" ref="jtaTransactionManager"/>

<property name="targetMethod" value="getTransactionManager"/>

</bean>-->

<!--<bean id="txManager" class="org.springframework.transaction.jta.JtaTransactionManager" >

</bean>-->

<bean id="bfrcDataSource" class="org.springframework.jndi.JndiObjectFactoryBean">

<property name="jndiName" value="java:/comp/env/jdbc/CrmData"/>

<property name="resourceRef" value="false" />

</bean>

<bean id="bfrcSessionFactory" class="org.springframework.orm.hibernate3.LocalSessionFactoryBean">

<property name="lobHandler">

<ref local="oracleLobHandler"/>

</property>

<property name="dataSource">

<ref local="bfrcDataSource" />

</property>

<property name="configLocation" value="classpath:/com/bfrc/framework/dao/hibernate3/hibernate.cfg.xml" />

<!--<property name="jtaTransactionManager">

<ref local="bfrcDataSource" />

</property>-->

<property name="hibernateProperties">

<value>

hibernate.show\_sql=${hibernate.show\_sql}

<!--hibernate.transaction.factory\_class=org.hibernate.transaction.JTATransactionFactory-->

hibernate.dialect=org.hibernate.dialect.Oracle10gDialect

</value>

</property>

</bean>

1. Added oracle jar to tomcat lib file
2. Added java-mail 1.4 jar to tomcat lib file
3. Removed server-config folder from web-inf and pasted in resource folder
4. Add jstl-1.2 jar in tomcat lib folder
5. In TLD files remove <body-content>JSP</body-content> and add <body-content>scriptless</body-content>
6. For Log file add the following entry in log4j.propertie file

log4j.rootLogger=INFO, file

#Redirect to Tomcat logs folder

log4j.appender.file.File=${catalina.home}/logs/Fcac.log

log4j.appender.file.MaxFileSize=10MB

log4j.appender.file.MaxBackupIndex=10

log4j.appender.file.layout=org.apache.log4j.PatternLayout

log4j.appender.file.layout.ConversionPattern=%d{yyyy-MM-dd HH:mm:ss} %-5p %c{1}:%L - %m%n

///////////////////////////////////////////////////

**TP\_DESKTOP**

1. For Data base connectivity add the below lines in the file **/ TiresPlus /src/main/webapp/WEB-INF/spring/daoContext.xml**

<bean id="bfrcDataSource" class="org.springframework.jndi.JndiObjectFactoryBean">

<property name="jndiName" value="java:/comp/env/jdbc/CrmData"/>

<property name="resourceRef" value="false" />

</bean>

1. Create a transaction Manager bean in the file **/ TiresPlus /src/main/webapp/WEB-INF/spring/daoContext.xml**

<bean id="txManager" class="org.springframework.orm.hibernate3.HibernateTransactionManager">

<property name="sessionFactory">

<ref local="sessionFactory" />

</property>

</bean>

1. To change oracle+websphere lob handler to tomcat default blob handler in **/ TiresPlus /src/main/webapp/WEB-INF/spring/daoContext.xml** file.

<!--<bean id="nativeJdbcExtractor" class="org.springframework.jdbc.support.nativejdbc.WebSphereNativeJdbcExtractor"/>-->

<bean id="nativeJdbcExtractor" class="org.springframework.jdbc.support.nativejdbc.CommonsDbcpNativeJdbcExtractor"/>

1. Comment the JTATransactionManager bean and replace all jtaTransactionManager to txManager in **/ TiresPlus /src/main/webapp/WEB-INF/spring/daoContext.xml** and **/TiresPlus/src/main/webapp/WEB-INF/applicationContext.xml**

<!-- JTA Transactions -->

<!--<bean id="jtaTransactionManager" class="org.springframework.transaction.jta.WebSphereUowTransactionManager" />

<bean id="txManager" class="org.springframework.beans.factory.config.MethodInvokingFactoryBean">

<property name="targetObject" ref="jtaTransactionManager"/>

<property name="targetMethod" value="getTransactionManager"/>

</bean>-->

And comment the hibernate properties in daoContext file

<!--hibernate.transaction.factory\_class=org.hibernate.transaction.JTATransactionFactory

hibernate.transaction.manager\_lookup\_class=org.hibernate.transaction.WebSphereTransactionManagerLookup-->

1. Add the Type cast in the file **/TiresPlus/src/main/webapp/WEB-INF/spring/siteContext.xml**

<property name="secureCall" value="#{new Boolean('${bing.maps.ajax.control.secure-call}')}" />

<property name="includeVersion" value="#{new Boolean('${bing.maps.ajax.control.include-version}')}" />

in Catalina.bat file you can find some text like below

set \_EXECJAVA=%\_RUNJAVA%

set MAINCLASS=org.apache.catalina.startup.Bootstrap

set ACTION=start

set SECURITY\_POLICY\_FILE=

set DEBUG\_OPTS=

set JPDA=

under that you need to put the

set CATALINA\_OPTS=-Xms512m -Xmx512m