**Installing TomEE Webapp Plus All In One Bundle**

* Download the apache-tomee-plus-1.7.3.zip , or apache-tomee-7.1.2-plus.zip file.
* Unzip the TomEE zip file.
* If not already created, create and set ***JAVA\_HOME*** to the **JDK** (not the JRE) location.
* Add TomEE as Windows service. Navigate to the <TomEE>/bin directory and execute the batch file, *service.install.as.admin.bat*, as an Administrator. The service description should appear as *Apache TomEE* in the services applet.
* To access a database within the context of TomEE installation, add *jconn3.jar* or the latest sajdbc4.jar to <TomEE>/lib. This will recognize the SQL Anywhere database connections.
* Add other .jar files to <TomEE>/lib in which your web applications may be dependent.
* Navigate to and edit <TomEE>/conf/tomee.xml file.
* Uncomment the line *<Deployment dir=”apps” />*
* To access database connections from within the context of one or more Message Driven Beans, add the following entries in between the <tomee> </tomee> tags, but above the line <Deployment dir=”apps” />:

*<Resource id = "jdbc/accounting" type="DataSource">*

*JdbcDriver com.sybase.jdbc3.jdbc.SybDriver*

*JdbcUrl jdbc:sybase:Tds:localhost:2638?ServiceName=accounting*

*UserName dba*

*Password sql*

*JtaManaged true*

*</Resource>*

*<Resource id = "jdbc/projecttracker" type="DataSource">*

*JdbcDriver com.sybase.jdbc3.jdbc.SybDriver*

*JdbcUrl jdbc:sybase:Tds:localhost:2638?ServiceName=projecttracker*

*UserName dba*

*Password sql*

*JtaManaged true*

*</Resource>*

*<Resource id = "jdbc/authentication" type="DataSource">*

*JdbcDriver com.sybase.jdbc3.jdbc.SybDriver*

*JdbcUrl jdbc:sybase:Tds:localhost:2638?ServiceName=authentication*

*UserName dba*

*Password sql*

*JtaManaged true*

*</Resource>*

*<Resource id = "jdbc/addressbook" type="DataSource">*

*JdbcDriver com.sybase.jdbc3.jdbc.SybDriver*

*JdbcUrl jdbc:sybase:Tds:localhost:2638?ServiceName=contacts*

*UserName dba*

*Password sql*

*JtaManaged true*

*</Resource>*

*<Resource id = "jdbc/mime" type="DataSource">*

*JdbcDriver com.sybase.jdbc3.jdbc.SybDriver*

*JdbcUrl jdbc:sybase:Tds:localhost:2638?ServiceName=mime*

*UserName dba*

*Password sql*

*JtaManaged true*

*</Resource>*

*<Resource id="MyJmsResourceAdapter" type="ActiveMQResourceAdapter">*

*# Do not start the embedded ActiveMQ broker*

*BrokerXmlConfig =*

*ServerUrl = tcp://localhost:61616*

*</Resource>*

*<Resource id="MyJmsConnectionFactory" type="javax.jms.ConnectionFactory">*

*ResourceAdapter = MyJmsResourceAdapter*

*</Resource>*

*<Container id="MyJmsMdbContainer" ctype="MESSAGE">*

*ResourceAdapter = MyJmsResourceAdapter*

*</Container>*

* Save and close <TomEE>/conf/tomee.xml file.
* Add subdirectory, *apps*, under <TomEE> installation.
* From the root OS directory, create the following directory structure

AppServer

config

data

mime

archive

in

out

log

test

mime

archive

in

out

* Copy the latest version of ***RMT2AppServerConfig.xml*** and ***log4j.properties*** to AppServer/config subdirectory from the web\_server\_configuration project.
* Verify that the Web Services Internal Server project/war file is named, server-internal-api
* Verify that the Web Services External Server project/war file is named, server-external-api
* Start Tomcat: (Windows) from the service applet or (Unix and Linux) <TomEE install dir>\bin\tomee.sh. Alternatively for all systems, attach the TomEE server to Eclipse IDE and start the server from within the IDE environment.
* Check the logs to verify database connections and other resources.
* Restart Tomcat.
* Your TomEE installation is now complete!

**Configuring DataSource to be Accessible Anywhere in Application Context**

1. Add DataSource declaration in context.xml for each application (.jar) as such:

<Resource name=*"jdbc/addressbook"*

auth=*"Container"*

type=*"javax.sql.DataSource"*

driverClassName=*"com.sybase.jdbc3.jdbc.SybDriver"*

url=*"jdbc:sybase:Tds:localhost:2638?ServiceName=contacts"*

username=*"contacts\_user"*

password=*"hoover"*

maxActive=*"10"* maxIdle=*"5"* maxWait=*"10000"*/>

Or add all declarations within the WEB-INF\resource.xml for the web application that accesses one or more standalone .jar files within the war file.

1. Add corresponding DataSource Ref in the web.xml

<resource-ref>

<description>Address Book DB Connection</description>

<res-ref-name>jdbc/addressbook</res-ref-name>

<res-type>javax.sql.DataSource</res-type>

<res-auth>Container</res-auth>

</resource-ref>