

Connect5 Server Configuration

Table of Contents

[Overview 3](#_Toc372621294)

[All Servers 3](#_Toc372621295)

[WebSphere 3](#_Toc372621296)

[Tomcat 3](#_Toc372621297)

[WebLogic 3](#_Toc372621298)

[JBoss 4](#_Toc372621299)

# Overview

This document outlines the differences between each major J2EE server and the customizations necessary to run Connect5.

# All Servers

For optimal JVM tuning, please apply the following parameters:

**-XX:+CMSClassUnloadingEnabled   
 -XX:+UseConcMarkSweepGC   
 -XX:MaxPermSize=256m   
 -XX:PermSize=128m**

# WebSphere

If you are using the File Upload Widget on WebSphere we found it had severe memory leaks due to the use of built in implementation of JAX-WS/SAAJ (AXIS2). To resolve this issue apply the following JVM parameters:

**-Djavax.xml.soap.MetaFactory=  
 com.sun.xml.internal.messaging.saaj.soap.SAAJMetaFactoryImpl   
-Djavax.xml.soap.SOAPConnectionFactory=  
 com.sun.xml.internal.messaging.saaj.client.p2p.HttpSOAPConnectionFactory   
-Djavax.xml.soap.MessageFactory=  
 com.sun.xml.internal.messaging.saaj.soap.ver1\_2.SOAPMessageFactory1\_2Impl**

# Tomcat

If you are using any polling listener that uses Quartz scheduler under the hood you will need an implementation of JTAs User Transaction. All J2EE containers have transaction support except for Tomcat. To resolve this issue add the **jta.jar** to your server’s lib directory.

# WebLogic

If you are using the File Upload Widget the default implementation of JAX-WS/SAAJ has several performance issues. To resolve this issue apply the following JVM parameters:

**-Djavax.xml.soap.MetaFactory=  
 com.sun.xml.internal.messaging.saaj.soap.SAAJMetaFactoryImpl   
-Djavax.xml.soap.SOAPConnectionFactory=  
 com.sun.xml.internal.messaging.saaj.client.p2p.HttpSOAPConnectionFactory   
-Djavax.xml.soap.MessageFactory=  
 com.sun.xml.internal.messaging.saaj.soap.ver1\_2.SOAPMessageFactory1\_2Impl**

In addition the above change we need to set the class loader policy to prefer Sun JARs. This achieved by including the **weblogic.xml** file in the WEB-INF directory of the WAR/EAR.  


If you continue to have problems you may need to add the latest SAAJ jars to WEB-INF/lib and remove ‘**internal**’ from the JVM parameters above.

# JBoss

Out of the box JBoss does not provide class loader isolation, AgencyPortal was getting transaction managerclass created by Connect5, which means any class level initialization specific to Connect5 was used by AgencyPortal. A **jboss-classloading.xml** and **jboss-web.xml** was added to the web app so that its gets its own domain and that solved the issue.

 

Add the following JARs to endorsed lib in JBoss:

**lib/endorsed/jbossws-native-jaxrpc.jar  
 lib/endorsed/jbossws-native-jaxws-ext.jar  
 lib/endorsed/jbossws-native-jaxws.jar  
 lib/endorsed/jbossws-native-saaj.jar**

Added the attribute in red below to **/jboss-5.1.0.GA/server/default/conf/bootstrap/Profile.xml**, otherwise JBoss won't deploy our apps in the JDK 1.6:

**<bean name="AttachmentStore" class="org.jboss.system.server.profileservice.repository.AbstractAttachmentStore">  
 <constructor><parameter class="java.io.File" ><inject bean="BootstrapProfileFactory" property="attachmentStoreRoot" /></parameter></constructor>  
 <property name="mainDeployer"><inject bean="MainDeployer" /></property>  
 <property name="serializer"><inject bean="AttachmentsSerializer" /></property>  
 <property name="persistenceFactory"><inject bean="PersistenceFactory" /></property>  
 </bean>**