
Technical Paper

Upgrading Web Application Servers That
Are Used with the SAS[®] 9.3 Middle-Tier
Configuration

Table of Contents

Overview	3
Motivation for Upgrade.....	3
Upgrading an Oracle WebLogic Application Server	4
Performing a Minor Upgrade of the WebLogic Application Server.....	4
WebLogic Installation Generates a <code>DemoIdentity</code> Certificate with a Different Common Name than Expected	5
Performing a Major Upgrade of the WebLogic Application Server.....	5
Upgrading an IBM WebSphere Application Server	6
Performing a Major Upgrade of a WebSphere Application Server.....	6
Upgrading a Red Hat JBoss Application Server (EAP Version).....	6
Performing a Minor Upgrade of a JBoss Application Server (EAP Version).....	7
Performing a Major Upgrade of a JBoss Application Server (EAP Version).....	7

Overview

This document provides an overview of the steps that are required to upgrade a web-application server installation and integrate those changes with an existing SAS® 9.3 middle-tier configuration. **Upgrading your web application server that you use with SAS 9.3 requires advanced knowledge of your configuration.**

The following application servers are supported and addressed in this guide:

- Red Hat JBoss Application Server (EAP)
- Oracle WebLogic Server
- IBM WebSphere Application Server

Important Notes:

To upgrade the Java Development Kit (JDK) that is in use with an existing SAS 9.3 middle-tier configuration, see the section "Java Development Kits and Java Application Servers" in "Configuring SAS® 9.3 to Use an Alternative Java Runtime Environment or Java Development Kit," available at support.sas.com/resources/thirdpartysupport/v93/AlternateJRE.html#appserverjdk.

To configure a new SAS 9.3 middle tier, see the SAS® 9.3 *Intelligence Platform: Installation and Configuration Guide*, available at support.sas.com/documentation/cdl/en/biig/62611/PDF/default/biig.pdf.

For information about migrating a SAS 9.2 configuration to a SAS 9.3 installation, see the SAS® 9.3 *Intelligence Platform: Migration Guide*, available at support.sas.com/documentation/cdl/en/bimig/62613/PDF/default/bimig.pdf.

Be sure to review "Third-Party Software Requirements for Use with SAS® Software and Applications" (support.sas.com/resources/thirdpartysupport/index.html) for minimum version requirements for SAS 9.3 software and any required patches or updates.

Motivation for Upgrade

When you initially configured your SAS 9.3 middle tier, you should have selected a Java Development Kit and web application server that meet the minimal requirements as documented in "Third-Party Software Requirements for Use with SAS® Software and Applications" (support.sas.com/resources/thirdpartysupport/index.html).

Periodically, third-party vendors release updates to address defects and security issues. Also, SAS Research and Development might discover issues with these third-party applications that motivate making a recommendation to upgrade in certain situations. As a policy, SAS supports a baseline release and minor updates in addition to that baseline. The support policy is available for review at support.sas.com/resources/thirdpartysupport/baseline_plus.html.

After you have determined that an application-server upgrade is appropriate for your SAS middle tier, follow the steps in the remaining sections to perform the upgrade.

Upgrading an Oracle WebLogic Application Server

SAS 9.3 supports Oracle WebLogic Server 10.3 and later. Support for Oracle WebLogic 12c was added in the second maintenance release for SAS 9.3 (TS1M2).

If you upgrade to Java 7, then you need to upgrade to Oracle WebLogic 12.1.1.

Performing a Minor Upgrade of the WebLogic Application Server

The following steps assume that you are upgrading your WebLogic Server from at least 10.3.0 to a later minor version (for example, from version 10.3.0 to 10.3.4). It is also assumed that you are using the **WebLogic Server Upgrade Installer** to update the existing WebLogic installation.

Before you begin, review the Oracle system requirements. The JDK version requirement might increase. Therefore, you might need to update the JDK before you upgrade the application server. If the new JDK is in a different location, review “Configuring SAS® 9.3 to Use an Alternative Java Runtime Environment or Java Development Kit” (support.sas.com/resources/thirdpartysupport/v93/AlternateJRE.html#appserverjdk) to make sure that SAS configuration tools know about the new location. For a minor version upgrade, it is recommended that you test a new JDK by starting the application server(s) and validating that SAS applications still work before introducing the application-server upgrade.

For information about required WebLogic Server patches, see “Oracle WebLogic Server Support for SAS® 9.3,” available at support.sas.com/resources/thirdpartysupport/v93/appservers/weblogicsw.html.

To perform a minor version upgrade of WebLogic 10.3.x, complete these steps:

1. Stop all of the WebLogic managed servers, the Administration Server, and the Node Manager.
2. Back up the WebLogic domain.
3. Remove all patches from the WebLogic Server installation using Oracle Smart Update. (download.oracle.com/docs/cd/E14759_01/doc.32/e14143/toc.htm).
4. Use the WebLogic Server Upgrade Installer to perform the upgrade. Download the upgrade installer from the “My Oracle Support” web page (support.oracle.com/CSP/ui/flash.html).
5. Apply any patches that are required for the version of WebLogic Server to which you are upgrading.
6. Restart the Node Manager, the Administration Server, and all of the managed servers.
7. Verify that SAS 9.3 web applications initialize successfully after the web-application server upgrade is complete.
8. Perform the site-specific configuration of customizations that apply to your configuration, if needed.

WebLogic Installation Generates a DemoIdentity Certificate with a Different Common Name than Expected

If you upgrade from version 10.3.0, 10.3.1, or 10.3.2 to version 10.3.3, the WebLogic Server Upgrade Installer generates the **DemoIdentity** certificate with a different common name (CN) than expected. This causes the **DemoIdentity** certificate to have the host name **rn360-a1-20**. The name on the **DemoIdentity** certificate must match the machine name. Otherwise, the WebLogic Configuration step fails. This security feature is known as Host Name Verification. It is used to avoid man-in-the-middle" security attacks, which is a rogue program attempt to spoof a server into thinking it is communicating with a different program.

After the WebLogic upgrade installation and before you use the SAS® Deployment Wizard, you must regenerate the **DemoIdentity** certificate and the **DemoIdentity** keystore using these steps:

1. From a command prompt, run one of the following scripts (based on your operating system):

Windows:

```
WebLogic-home-directory\server\bin\setWLSEnv.cmd
```

UNIX: (use the 'dot' command to run the script in the same shell)

```
WebLogic-home-directory/server/bin/setWLSEnv.sh
```

2. Then, execute this command:

```
java utils.CertGen -keyfilepass DemoIdentityPassPhrase
                  -certfile democert -keyfile demokey
```

3. Finally, execute this command:

```
java utils.ImportPrivateKey -keystore DemoIdentity.jks -storepass
DemoIdentityKeyStorePassPhrase -keyfile demoKey -keyfilepass
DemoIdentityPassPhrase -certfile democert.pem -keyfile demokey.pem
                        -alias demoidentity
```

4. Replace *WebLogic-home-directory*\server\lib\DemoIdentity.jks with the file that is generated in step3.

Performing a Major Upgrade of the WebLogic Application Server

Because WebLogic Server 12.1.1 is a new major release, you must install the new software (that is, there is no upgrade installer). However, Oracle does provide tools to help upgrade your web-application environment. For information about upgrading your environment from WebLogic 10.3.x to WebLogic Server 12.1.1, see to the "Oracle® Fusion Middleware Upgrade Guide for Oracle WebLogic Server 12c Release 1 (12.1.1)", available at docs.oracle.com/cd/E24329_01/web.1211/e24497/toc.htm.

Upgrading an IBM WebSphere Application Server

SAS 9.3 TS1M0 and TS1M1 support IBM WebSphere Application Server Version 7.0. Support for IBM WebSphere Version 8.0. and 8.5 is available in SAS 9.3 TS1M2.

Performing a Major Upgrade of a WebSphere Application Server

To perform a major version upgrade of IBM WebSphere Application Server from version 7.0 to version 8.0 or version 8.5, follow these steps:

1. Install IBM WebSphere Application Server Version 8.0 or 8.5 using the IBM WebSphere Application Server Information Center documentation, available at either of the following web sites:
 - pic.dhe.ibm.com/infocenter/wasinfo/v8r0/index.jsp
 - pic.dhe.ibm.com/infocenter/wasinfo/v8r5/index.jsp
2. Download the IBM WebSphere Application Server Version 8.0 or 8.5 Fix Pack. Then, use the IBM Installation Manager to apply the fix-pack maintenance to bring it to a minimum level of 8.0.0.2 or 8.5.0.1.
3. Use the IBM WebSphere Application Server Version 8.0 or 8.5 Migration Wizard or Migration Tools to migrate the existing WebSphere Application Server Version 7.0 profiles and SAS web applications to either the Web Application Server Version 8.0 or the 8.5 environment.
4. Update the WebSphere location that is saved in the SAS Deployment Registry by submitting a Java command with the following syntax:

```
java -jar "SASHOME/deploymntreg/sas.tools.deploymntreg.jar"  
    setInstallLoc Webspherend-12byte-name default  
    "WebSphere-installation-path"
```

For example, the following command upgrades to WebSphere 8.0:

```
java -jar "C:/Program Files/SAS/deploymntreg/sas.tools.deploymntreg.jar"  
    setInstallLoc webspherend default  
    "C:\Program Files\IBM\WebSphere_v8.0\AppServer"
```

5. Restart the deployment manager, the node agent, and all WebSphere servers.
6. Verify that SAS 9.3 web applications initialize successfully after the web-application server upgrade is complete.
7. Perform a site-specific configuration of customizations that apply to your configuration, if necessary.

Upgrading a Red Hat JBoss Application Server (EAP Version)

SAS 9.3 only supports JBoss 4.3 using JDK 6, JBoss 5.1, and JBoss 5.2 using JDK 7. Using JDK 6, JBoss 4.3 consists of a set of component JAR files that have unique version numbers. Any upgrades apply to individual JAR files. If necessary, follow the SAS Baseline and Higher Policy and upgrade the individual JBoss 4.3 JAR files.

If you upgrade to Java 7, then you need to upgrade to Red Hat JBoss EAP 5.2.0 or JBoss Community v7.1.1

You can perform upgrades to JBoss either before or after completing the SAS maintenance upgrade.

Performing a Minor Upgrade of a JBoss Application Server (EAP Version)

To perform an EAP version minor upgrade of JBoss 4.3, JBoss 5.1 or JBoss 5.2 for SAS 9.3, complete these steps:

1. Stop all JBoss servers.
2. Stop all SAS servers in this sequence: the SAS® Deployment Tester server, SAS Services Application (Remote Services), SAS® Table Server, SAS/CONNECT® spawner, SAS/SHARE® server, SAS object spawner, SAS® OLAP Server, and the SAS® Metadata Server.
3. Upgrade to the later versions of the JBoss JAR files.
4. Restart all SAS and JBoss servers.
5. Verify that SAS 9.3 web applications initialize successfully after the web-application server upgrade is complete.
6. Perform a site-specific configuration of customizations that apply to your configuration, if necessary.

Performing a Major Upgrade of a JBoss Application Server (EAP Version)

Note: Before you perform the following instructions, ensure that you are running SAS® 9.3 software.

1. Make a backup of your entire system (all tiers). **DO NOT CONTINUE** without a backup.
2. Install the new version of JBoss (for example, 5.1 or 5.2).
3. Stop the old version of JBoss (for example, 4.3) and SAS Remote Services.
4. Update the JBoss and JDK locations that are saved in the SAS Deployment Registry by submitting Java commands with the following syntax:

```
java -jar $SASHOME/deploymntreg/sas.tools.deploymntreg.jar
      setInstallLoc jboss default path-to-JBoss
```

```
java -jar $SASHOME/deploymntreg/sas.tools.deploymntreg.jar
      setInstallLoc jdk default path-to-JDK
```

Sample Commands for upgrade to JBoss 5.1:

```
java -jar "C:/Program
          Files/SASHome/deploymntreg/sas.tools.deploymntreg.jar"
      setInstallLoc jboss default "C:/jboss-eap-5.1/jboss-as"
```

```
java -jar "C:/Program
          Files/SASHome/deploymntreg/sas.tools.deploymntreg.jar"
      setInstallLoc jdk default "C:/Java/jdk1.6.0_30"
```

Sample Commands for upgrade to JBoss 5.2:

```
java -jar "C:/Program
    Files/SASHome/deploymntreg/sas.tools.deploymntreg.jar"
    setInstallLoc jboss default "C:/jboss-eap-5.2/jboss-as"

java -jar "C:/Program
    Files/SASHome/deploymntreg/sas.tools.deploymntreg.jar"
    setInstallLoc jdk default "C:/Java/jdk1.7.0_15"
```

5. For backup purposes, copy the directory *SAS-configuration-directory/Lev1/Web/Scripts/JBoss* to the directory *SAS-configuration-directory/Lev1/Web/Scripts/JBossx.x*.
6. Set the JBOSS_HOME environment variable to the location of the new 5.x version. This variable is located in the jbossScripting.bat file (for Windows environments) or in the jbossScripting.sh file (for UNIX environments). Both of these files are located in the *SAS-configuration-directory/Lev1/Web/Scripts/JBoss* directory for the respective operating environments.

For example, suppose you have the following settings for the JBOSS_HOME variable:

```
set JBOSS_HOME=C:\jboss-4.3
set JBOSS_VERSION=4
```

If you are upgrading to JBoss 5.2, then change the settings as follows:

```
set JBOSS_HOME=C:\jboss-eap-5.2\jboss-as
set JBOSS_VERSION=5
```

7. Modify the jbossScripting.properties file, located in *SAS-configuration-directory/Lev1/Web/Scripts/JBoss*, as follows:
 - a. Change the `config.appserver.version=` parameter to the updated JBoss version.

For example, suppose the current setting is the following:

```
config.appserver.verson=4.3 (build\SVNTag\=JBoss_4_2_3_GA
date\=200807181439)
```

For JBoss 5.1, change that setting as follows:

```
config.appserver.version=5.1 (build\SVNTag\=JBPAPP_5_1_0
date\=201009150028)
```

- b. Globally replace references to the JBoss installation directory with the new JBoss directory name.
 - c. Change the Java Virtual Memory (JVM) options to the values that are shown below in bold:

```
server.server1.options=-Xms1024m; -Xmx3096m; -XX\:PermSize\=768m;
    -XX\:MaxPermSize\=768m; -XX
    . . .more parameters. . .
```

8. Verify that the following entries are in the `jbossScripting.properties` file. If these entries are not in the file, add them before you run the scripts in the next step.

```

queue=queue1 queue2 queue3 queue4 queue5 queue6 queue7 queue8
queue.queue1.jndiname=sas/jms/AlertQueue
queue.queue1.name=AlertQueue
queue.queue1.servername=SASServer1
queue.queue2.jndiname=sas/jms/WorkflowQueue
queue.queue2.name=WorkflowQueue
queue.queue2.servername=SASServer1
queue.queue3.jndiname=sas/jms/AuditQueue
queue.queue3.name=AuditQueue
queue.queue3.servername=SASServer1
queue.queue4.jndiname=sas/jms/WorkflowCommandQueue
queue.queue4.name=WorkflowCommandQueue
queue.queue4.servername=SASServer1
queue.queue5.jndiname=sas/jms/app/WorkflowEventsQueue
queue.queue5.name=WorkflowEventsQueue
queue.queue5.servername=SASServer1
queue.queue6.jndiname=sas/jms/scheduler.pip.resultsQueue
queue.queue6.name=scheduler.pip.resultsQueue
queue.queue6.servername=SASServer1
queue.queue7.jndiname=sas/jms/scheduler.pip.outputQueue
queue.queue7.name=scheduler.pip.outputQueue
queue.queue7.servername=SASServer1
queue.queue8.jndiname=sas/jms/scheduler.pip.jobQueue
queue.queue8.name=scheduler.pip.jobQueue
queue.queue8.servername=SASServer1
topic=topic1 topic2 topic3
topic.topic1.jndiname=sas/jms/NotificationEventTopic
topic.topic1.name=NotificationEventTopic
topic.topic1.servername=SASServer1
topic.topic2.jndiname=sas/jms/SASPublishedEventTopic
topic.topic2.name=SASPublishedEventTopic
topic.topic2.servername=SASServer1
topic.topic3.jndiname=sas/jms/scheduler.pip.broadcastTopic
topic.topic3.name=scheduler.pip.broadcastTopic
topic.topic3.servername=SASServer1
connectionfactory=connectionfactory1 connectionfactory2
connectionfactory.connectionfactory1.jndiname=sas/jms/TopicConnectionFactory
connectionfactory.connectionfactory1.name=SASTopicConnectionFactory
connectionfactory.connectionfactory1.servername=SASServer1
connectionfactory.connectionfactory2.jndiname=sas/jms/QueueConnectionFactory
connectionfactory.connectionfactory2.name=SASQueueConnectionFactory
connectionfactory.connectionfactory2.servername=SASServer1

```

9. Before you run the script file (`jbossScripting.bat` for Windows or `jbossScripting.sh` for UNIX), replace the one occurrence of `groovy-all-1.5.1.jar` in that file with `groovy-all.jar`.
10. Run the script file (either `jbossScripting.sh` or `jbossScripting.bat`) that is located in *SAS-configuration-directory/Lev1/Web/Scripts/JBoss*.
11. Restart all SAS and JBoss servers.
12. Verify that SAS 9.3 web applications initialize successfully after the web-application server upgrade is complete.
13. Perform a site-specific configuration of customizations that apply to your configuration, if necessary.

