# **Stack 6 JBoss - Windows Runbook**

The **Stack 6 for JBoss - Windows Runbook** shows how to configure TAFJ-Transact on JBoss EAP 7.3 with SQL Server 2019 database. It also shows you how to deploy BrowserWeb and UXP Browser and access Transact using these browsers. This runbook is relevant to all post R20 AMR releases up to and including R21 AMR.

**NOTE:**

This runbook doesn't tell you how to install third-party software. For more information see the relevant vendor's documentation.

## Stacks table

For the latest version of the stacks table, see either the Temenos Customer Portal or the Temenos Partner Portal.

### Scope

This runbook covers:

* Installing TAFJ.
* Installing Transact.
* Configuring JBoss EAP 7.3.
* Deploying Transact and TAFJ artefacts in JBoss.

## Audience

This runbook is designed for technical consultants who want to deploy TAFJ and Transact on a JBoss application server and access it through a browser.

## Skills and knowledge

To use this runbook, you need a basic understanding of the following:

* Transact
* TAFJ
* JBoss
* MS SQL Server

# Introduction

This runbook helps you to set up the development or test environment for Transact and TAFJ on the Windows operating system. It shows you how to install and configure TAFJ, Transact and JBoss and how to access Transact through both BrowserWeb and UXP Browser. In this exercise, a two-tier architecture has been created which consists of an application server and a database server.

## Environment

The application tier contains JBoss EAP (version 7.3) application server with TAFJ and Transact. The database tier contains the MS SQL Server 2019 database that stores Transact data. For better understanding, the architecture is given below.

[](http://172.16.204.9/R21UG/Solutions/Runbooks/Stack-06-Runbook-R21/img/architecture.svg)

## UXP Browser Generate On Demand (GOD) mode

In releases prior to R20, there was a large directory in UXP Browser on the web server file system (BRP\generated) which contained over 250,000 IRIS and Edge files. To manage such a large number of files securely and keep the system scalable was difficult.

In R20 and later releases, model-derived artefacts are no longer pre-generated. Instead, they are generated on-demand and saved in the Resource Server database. As a result, the start-up time of the system has been significantly reduced.

The first time you access a screen, there is a one-off delay as the system generates the screen. Once this completes, the screen is stored for all subsequent requests. The generated screens are stored in the database permanently until the underlying definition changes in Transact.

## Components deployed in the GOD mode of UXP Browser

You must configure and deploy the following components to deploy and run UXP Browser in the GOD mode.

**Authenticator (Authenticator.war)**

This component validates Transact user credentials. It authenticates user names and passwords against Transact using the tComponent framework and ensures there is secure communication between Browser and IRIS.

**UXP Browser (Browser.war)**

This component renders the new UXP Browser user interface. It communicates with the Resource Server to render front end screens and with the IRIS layer to use Transact data from the database. Its main benefits are: better scalability, the SaaS capability and support for multi tenancy.

**Resource server (ResourceServer.war)**

This component uses a database schema (within each per-tenant Transact database) providing an alternative to the file system for the storage and retrieval of tenant-specific variants of resources that are external to Browser.war, for example artefacts generated from models (projects, menus, templates etc.) and static resources (CSS stylesheets, HTML files, etc.).

**Interaction Framework Resource Provider service (irf-rp-services.war)**

This service processes requests and responses through the Resource Provider (RP) component apart from through traditional Open Financial Service (OFS).

Using the Interaction Framework Resource Provider service, you can communicate with Transact to perform various Transact operations such as create, view, delete, update and so on, for any versions and fetch any enquiry results without any hassle of metadata generation.

**Interaction Framework Catalog service (irf-t24catalog-services.war)**

This service was designed in a generic way to retrieve metadata information for any Transact artefact in an XML formatted response. The Interaction Framework Catalog service is used by UXP Browser deployed in the Generate On Demand mode as well as Design Studio on DSF Packager.

**Enterprise Java Beans (EJB files)**

The EJBs are packaged as JAR files and are deployed on the Transact application server to support remote EJB connectivity. They enable APIs and Transact running on separate or remote application servers to communicate with each other. The following EJBs are required:

* Authentication service EJB (t24-EB\_AuthenticationService-ejb.jar)
* Catalog service EJB (t24-EB\_CatalogService-ejb.jar)
* Resource Provider service EJB (t24-EB\_ResourceProviderService-ejb)

# Prerequisites

## Software prerequisites

You must first obtain all the required artefacts from your account manager. Before you start installing and configuring the stack, you must install all third-party software.

### Third-party software

| Software | Version |
| --- | --- |
| Windows server | 2019 |
| JDK | 1.8 |
| JBoss | 7.3 |
| Microsoft SQL Server | 2019 |
| SQL Server Management Studio | 2019 |
| Axis2 | 1.6.2 |

### Temenos artefacts

| Artefacts | Files | Description |
| --- | --- | --- |
| Database backup | MB.202008.MSSQL\_2014.WIN.TAFJ202008.30-SEP-2021.bak.tar.gz | Contains the database. |
| Transact | MB.202008.TAFJ202008.bnk.tar.gz | Contains the bnk directory that holds the Transact libraries. |
| TAFJ | TAFJ.DEV.202008.1.tar.gz | Contains the TAFJ runtime JAR file, TAFJ patch script and TAFJ setup script. |
| Transact Browser | BrowserWeb-202008.00.tar | Old browser components. |
| UXP Browser | UXP-Browser.202008.zip  The following required files are embedded in the package:   * Authenticator-202008.0.1.war * Browser-202008.0.1.war * irf-rp-services-202008.0.1.war * ResourceServerWar-202008.0.1.war * BRPRuntimeProperties.properties * RSDefaultProperties.properties | New browser components. |
| Interaction Framework Catalog services | irf-t24catalog-services-202008.0.1.war | Catalog services WAR file |
| Additional EJB JAR files | t24-EB\_AuthenticationService-ejb.jar  t24-EB\_CatalogService-ejb.jar  t24-EB\_ResourceProviderService-ejb.jar  These files are available in the subfolders under T24\bnk\Extensions. | EJB JARs required for deploying UXP Browser in Generate On Demand (GOD) mode |
| Temenos Encryption Utility | EncryptPassword.jar | The utility for encrypting passwords. It is required for configuring UXP in GOD mode. |

# Before you start

Before you start installing and configuring the stack, you need to check that your technical infrastructure has been set up properly.

**NOTE:**

This runbook does not tell you how to install third-party software. For more information, see the relevant vendor's documentation

## Completing preinstallation tasks

**Procedure**

1. Load Transact data into the MS SQL Server from the .bak file.

**T24 DataBase Restore**

1.Copy the  R20 MSSQL file from bucket and uncompress

Copy command : gsutil cp gs://temenos-dev-backup-bucket/R20/MB.R20.MSSQL\_2014.WIN.TAFJR20.Training.31-OCT-2020.bak.tar.gz c:\R19MB\.

🡪uncompress the file from tar.gz to .tar and then uncompress the .tar to .bak file.

2.Open Microsoft SQL Server management Studio.

A computer screen with a computer screen

Description automatically generated

3.Connect to dev-enviroment server by selecting from server name drop down menu

4.Databases>RestoreDatabase

A screenshot of a computer

Description automatically generated

1. Load TAFJ-related stored procedures and Java functions into the database. For more information, see the database-specific installation document that is stored in %TAFJ\_HOME%\doc.
2. Install the required third-party software such as JDK and JBoss. For more information about the installation paths, see [**Creating the directory structure**](http://172.16.204.9/R21UG/Solutions/Runbooks/Stack-06-Runbook-R21/before-you-start.htm?tocpath=_____4#Creating).
3. Prepare all Temenos artefacts in the chosen temporary directory on your machine.
4. If you are going to use a non-standard port on the application server, ensure that you open it in the Windows firewall.

## Checking the operating system

**Procedure**

1. Display the Windows Control Panel.
2. Select **System and Security**.
3. Click **System**.
4. Check that the operating system is Windows Server 2019.

[](http://172.16.204.9/R21UG/Solutions/Runbooks/Stack-06-Runbook-R21/img/4.1-02.png)

## Checking the disk space

**Procedure**

1. In the File Explorer, right-click the C drive.
2. Click **Properties**. The Properties dialog is displayed.
3. Check that there is enough free space on your disk drive to install the software. The minimum requirement is 20 GB.

[](http://172.16.204.9/R21UG/Solutions/Runbooks/Stack-06-Runbook-R21/img/4.2-01.png)

## Creating the directory structure

In your user home directory, create the following directory structure with Temenos as the parent directory.

C:.

|

+---3rdParty

| +---AS

| +---JAVA

+---Install

+---T24

+---TAFJ

+---UXPBrowser

The following table shows the directory structure:

| Folder | Description |
| --- | --- |
| 3rdParty | All installed third-party software (e.g. JDK 1.8 and JBoss). |
| Install | All installers required to set up TAFJ and Transact. |
| T24 | All Transact-related libraries. |
| TAFJ | TAFJ runtime libraries. |
| UXPBrowser | Artefacts for the new UXP Browser. |

## Configuring the environment variables

**Procedure**

1. In the File Explorer, right-click **This PC**.
2. Click **Properties**. A dialog box is displayed.
3. Click **Advanced system settings**. Another dialog box is displayed.
4. Click **Environment Variables**.
5. Use the **New** button to add the following to the system variables.
   * BRP\_HOME=%TEMENOS\_HOME%
   * JAVA\_HOME=%TEMENOS\_HOME%\3rdParty\JDK
   * JBOSS\_HOME=%TEMENOS\_HOME%\3rdParty\AS\jboss-eap-7.3
   * T24\_HOME=%TEMENOS\_HOME%\T24\bnk\UD
   * TAFJ\_HOME=%TEMENOS\_HOME%\TAFJ
   * TEMENOS\_HOME=C:\Temenos
   * UXPBROWSER=%TEMENOS\_HOME%\UXPBrowser

[](http://172.16.204.9/R21UG/Solutions/Runbooks/Stack-06-Runbook-R21/img/4.4-01.png)

## Verifying the system for third-party software

**Procedure**

1. To verify that JDK 1.8 is installed, at the command prompt enter java -version. Add the **JAVA\_HOME** variable and %JAVA\_HOME%\bin to the %PATH% variable. You should see the Java version displayed.

[](http://172.16.204.9/R21UG/Solutions/Runbooks/Stack-06-Runbook-R21/img/4.5-01.png)

1. To check that you can connect to the SQL database, use SQL Server Management Studio (SSMS) to establish a connection.

[](http://172.16.204.9/R21UG/Solutions/Runbooks/Stack-06-Runbook-R21/img/4.5-02.png)

1. To verify that JBoss EAP 7.3 is installed, and that **JBOSS\_HOME** has been added as an environment variable, check that the following folders are present in %JBOSS\_HOME%.

[](http://172.16.204.9/R21UG/Solutions/Runbooks/Stack-06-Runbook-R21/img/4.5-04.png)

**Installing Database**

Database Installation guides documents are provided below:



Note:- For R22 release onwards, we are supposed to run the DBscritps as per the below attached document.



# Installing TAFJ runtime

## Extracting Transact

**Procedure**

1. Upload the Transact installation package (MB.202008.TAFJ202008.bnk.tar.gz) from the temporary folder on your machine to the %TEMENOS\_HOME%\install folder on the application server.
2. Navigate to %TEMENOS\_HOME%\install.
3. Extract the TAR file to the T24 directory (C:\Temenos\T24).

[](http://172.16.204.9/R21UG/Solutions/Runbooks/Stack-06-Runbook-R21/img/5.1-00.png)

The file is extracted as a bnk folder.

[](http://172.16.204.9/R21UG/Solutions/Runbooks/Stack-06-Runbook-R21/img/5.1-01.png)

## Installing TAFJ

**Note:**

Ensure that you have set the correct path for JAVA\_HOME

**Procedure**

1. Upload the TAFJ installation package (TAFJ.DEV.202008.1.tar.gz) from the temporary folder on your machine to the %TEMENOS\_HOME%\install folder on the application server.
2. In %TEMENOS\_HOME%\install, extract TAFJ.DEV.202008.1.tar.gz.

[](http://172.16.204.9/R21UG/Solutions/Runbooks/Stack-06-Runbook-R21/img/5.2-01.png)

1. In the %TEMENOS\_HOME%\install folder, open the command prompt.
2. Execute Setup\_TAFJ.DEV.202008.1.

[](http://172.16.204.9/R21UG/Solutions/Runbooks/Stack-06-Runbook-R21/img/5.2-04.png)

1. During the installation, when prompted for the installation directory, enter the path to %TAFJ\_HOME%.

[](http://172.16.204.9/R21UG/Solutions/Runbooks/Stack-06-Runbook-R21/img/5.2-05.png)

1. Specify y to allow the installer to create the TAFJ directory and press Enter.

[](http://172.16.204.9/R21UG/Solutions/Runbooks/Stack-06-Runbook-R21/img/5.2-05b.png)

1. Press **Enter** at the next prompt for Eclipse home directory.

[](http://172.16.204.9/R21UG/Solutions/Runbooks/Stack-06-Runbook-R21/img/5.2-05c.png)

1. Enter y to create the conf directory under %TAFJ\_HOME%.

[](http://172.16.204.9/R21UG/Solutions/Runbooks/Stack-06-Runbook-R21/img/5.2-06.png)

1. Press **Enter** to terminate. This completes the TAFJ  runtime installation.

[](http://172.16.204.9/R21UG/Solutions/Runbooks/Stack-06-Runbook-R21/img/5.2-07.png)

You should see the following folders in %TAFJ\_HOME%:

[](http://172.16.204.9/R21UG/Solutions/Runbooks/Stack-06-Runbook-R21/img/5.2-08.png)

1. In the File Explorer, open the command prompt from %TAFJ\_HOME%\bin. Execute the tVersion command to verify the TAFJ version.

[](http://172.16.204.9/R21UG/Solutions/Runbooks/Stack-06-Runbook-R21/img/5.2-09.png)

1. In Windows Defender restore the quarantined TemenosSecurity.jar.

**Note:**

Machines with security policy enabled under Windows Defender or Norton Antivirus might face some issues due to TemenosSecurity.jar, which is available as part of the TAFJ release. During the installation, you may see a warning message.

[](http://172.16.204.9/R21UG/Solutions/Runbooks/Stack-06-Runbook-R21/img/restore-Temenos-security.png)

## Deploying the database driver

TAFJ uses JDBC drivers to establish a connection with the database.

Database drivers are available in the %TAFJ\_HOME%\dbdrivers folder. For your deployment, you must use the latest version of the respective database driver downloaded from the database vendor’s portal.

**Procedure**

1. Download the JDBC SQL driver from Microsoft portal.
2. Copy sqljdbc42.jar to the %TAFJ\_HOME%\ext folder.

[](http://172.16.204.9/R21UG/Solutions/Runbooks/Stack-06-Runbook-R21/img/5.3-01.png)

 Dark Mode

[**Expand/Collapse**](http://172.16.204.9/R21UG/Solutions/Runbooks/Stack-06-Runbook-R21/install-tafj-runtime.htm?tocpath=_____5)

[**Share**](http://172.16.204.9/R21UG/Solutions/Runbooks/Stack-06-Runbook-R21/install-tafj-runtime.htm?tocpath=_____5)

[**Print**](http://172.16.204.9/R21UG/Solutions/Runbooks/Stack-06-Runbook-R21/install-tafj-runtime.htm?tocpath=_____5)

In this topic

* [**Installing TAFJ runtime**](http://172.16.204.9/R21UG/Solutions/Runbooks/Stack-06-Runbook-R21/install-tafj-runtime.htm?tocpath=_____5#InstallingTAFJruntime)
  + [Extracting Transact](http://172.16.204.9/R21UG/Solutions/Runbooks/Stack-06-Runbook-R21/install-tafj-runtime.htm?tocpath=_____5#ExtractingTransact)
  + [Installing TAFJ](http://172.16.204.9/R21UG/Solutions/Runbooks/Stack-06-Runbook-R21/install-tafj-runtime.htm?tocpath=_____5#InstallingTAFJ)
  + [Deploying the database driver](http://172.16.204.9/R21UG/Solutions/Runbooks/Stack-06-Runbook-R21/install-tafj-runtime.htm?tocpath=_____5#Deployingthedatabasedriver)

Copyright © 2020-202

# Configuring TAFJ

This section shows how to configure TAFJ runtime to set up Transact in standalone mode (i.e. outside an application server). TAFJ does not use environment variables. Instead, it uses a properties file (\*.properties) for its internal configuration. Properties files are stored under the %TAFJ\_HOME%\conf directory. The default property file is tafj.properties.

## Configuring tafj.properties

**Note:**

For more information on tafj.properties, see the the guides in %TAFJ\_HOME%\doc.

**Procedure**

1. Go to the %TAFJ\_HOME%\conf folder.
2. Open tafj.properties.
3. Ensure that tafj.home points to the correct location.

[](http://172.16.204.9/R21UG/Solutions/Runbooks/Stack-06-Runbook-R21/img/6.1-01.png)

1. Ensure that temn.tafj.directory.precompile points to the Transact libraries.

[](http://172.16.204.9/R21UG/Solutions/Runbooks/Stack-06-Runbook-R21/img/6.1-02.png)

1. Set up the database connection.

[](http://172.16.204.9/R21UG/Solutions/Runbooks/Stack-06-Runbook-R21/img/6.1-03.png)

* 1. Under the **Database setup** header section, enter a value for temn.tafj.jdbc.url.
  2. In temn.tafj.jdbc.url change integratedSecurity to false.
  3. Provide the relevant driver information in temn.tafj.jdbc.driver.
  4. Enter the database user name and password.

1. Ensure that temn.tafj.runtime.directory.current points to the UD folder in %T24\_HOME%.

[](http://172.16.204.9/R21UG/Solutions/Runbooks/Stack-06-Runbook-R21/img/6.1-04.png)

1. Set the value of temn.tafj.jdbc.write.use.merge and temn.tafj.jdbc.write.use.merge.no.xml to false (to improve database performance).

[](http://172.16.204.9/R21UG/Solutions/Runbooks/Stack-06-Runbook-R21/img/6.1-05.png)

1. Set the value of temn.tafj.locking.mode to DATABASE.

[](http://172.16.204.9/R21UG/Solutions/Runbooks/Stack-06-Runbook-R21/img/6.1-06.png)

This runbook uses DATABASE locking mode, which is recommended for releases 201809 onwards. TAFJ locks records using the native database locks when this locking mode is used.

**Note:**

See the TAFJ-Lock Manager user guide (TAFJ-Lock Manager.pdf) that is in %TAFJ\_HOME%\doc for information on creating and populating the TAFJ\_HASHLOCKS table (which is required for DATABASE locking mode).

## Verifying the TAFJ installation

**Procedure**

1. In %TAFJ\_HOME%\bin, open the command prompt and enter tDiag.bat. This will display the following information.
   * Where %TAFJ\_HOME% is set to.
   * The JDK version.
   * The TAFJ version.
   * The default project, and details of all the projects set in %TAFJ\_HOME%\conf.
2. Check that these values are correct.

An example of tDiag output is illustrated below.

[](http://172.16.204.9/R21UG/Solutions/Runbooks/Stack-06-Runbook-R21/img/6.1.1-01.png)

[](http://172.16.204.9/R21UG/Solutions/Runbooks/Stack-06-Runbook-R21/img/6.1.1-02.png)

# Sanity check

Before you deploy the application server, you must perform two sanity checks to verify that the Transact standalone installation was successful.

## Accessing Transact Classic

Classic is one of the Transact user interfaces.

**Procedure**

1. In the File Explorer, open the command prompt from %TAFJ\_HOME%\bin.
2. Enter tRun EX. The Classic command line interface is displayed.

[](http://172.16.204.9/R21UG/Solutions/Runbooks/Stack-06-Runbook-R21/img/7.1-00.png)

1. Specify a valid Transact user name and password and press Enter.

[](http://172.16.204.9/R21UG/Solutions/Runbooks/Stack-06-Runbook-R21/img/7.1-01.png)

1. Launch any Transact application, for example SPF S SYSTEM.

[](http://172.16.204.9/R21UG/Solutions/Runbooks/Stack-06-Runbook-R21/img/7.1-02.png)

## Accessing DBTools

DBTools is a TAFJ tool that provides a console to execute database commands. The console can only be accessed by authenticated users.

**Procedure**

1. In the File Explorer, go to %TAFJ\_HOME%\bin.
2. Create a TAFJ user by passing username and password parameters to the tUserMgnt.bat utility.

tUserMgnt --Add -u tafjuser -p Temenos@123

The USER CREATION SUCCESSFUL message is displayed.

1. Launch DBTools, passing the TAFJ user credentials as parameters.

DBTools -u tafjuser -p Temenos@123

The DBTools console is launched.

1. View the database name, database user name and IP address displayed in the bottom left of the screen.

[](http://172.16.204.9/R21UG/Solutions/Runbooks/Stack-06-Runbook-R21/img/7.2-02.png)

 Dark Mode

[**Expand/Collapse**](http://172.16.204.9/R21UG/Solutions/Runbooks/Stack-06-Runbook-R21/sanity-check.htm?tocpath=_____7)

[**Share**](http://172.16.204.9/R21UG/Solutions/Runbooks/Stack-06-Runbook-R21/sanity-check.htm?tocpath=_____7)

[**Print**](http://172.16.204.9/R21UG/Solutions/Runbooks/Stack-06-Runbook-R21/sanity-check.htm?tocpath=_____7)

# Configuring JBoss EAP 7.3

This section shows how to configure JBoss modules for TAFJ, Transact and the database.

## Configuration prerequisites

Before you start configuring JBoss with TAFJ and Transact, you need to make sure that:

1. TAFJ is installed (%TAFJ\_HOME%).
2. Transact precompiled JAR files are in the correct location (%T24\_HOME%\t24lib).
3. Your standalone mode (Classic) is working.

## Configuring memory settings

The default memory settings are as follows.

set "JAVA\_OPTS=-Xms1G -Xmx1G -XX:MetaspaceSize=96M -XX:MaxMetaspaceSize=256m"

You need to consider refining your memory setting based on your application server load and system RAM.

**Procedure**

1. Change the memory setting in the standalone.conf.bat file in *%JBOSS\_HOME%*\bin.

[](http://172.16.204.9/R21UG/Solutions/Runbooks/Stack-06-Runbook-R21/img/8.1.1-01.png)

**Note:**

When setting up the max heap size (mx), set the initial heap size (ms) to the same value as the max heap size. This prevents the JVM from consuming cycles to expand the heap since Transact starts with a large memory footprint.

1. Set the mandatory properties tafj.home and file.encoding in standalone.conf.bat. Note that TAFJ uses UTF-8 encoding.

[](http://172.16.204.9/R21UG/Solutions/Runbooks/Stack-06-Runbook-R21/img/8.1.1-02.png)

## Configuring JBoss modules

JBoss EAP 7.3 uses a modular class loading system for controlling the class paths of deployed applications. The modular class loader separates all Java classes into logical groups called modules. Each module can define dependencies on other modules.

Installing a module on JBoss EAP 7.3 requires creating a path under the %JBOSS\_HOME%\modules folder. Under this path, install the JAR libraries that are part of the module and a module.xml file that describes the module itself and dependencies with other modules. You need to create three modules on your application server.

* MS SQL Server database driver module.
* TAFJ module.
* Transact module.

### Deploying the MS SQL Server database driver module

For Transact to connect to a data source, this module must contain your data source vendors and JDBC drivers so they can be used by JBoss.

**Procedure**

1. In the File Explorer, go to the modules folder in JBoss (%JBOSS\_HOME%\modules).
2. In the modules folder, create the following directory structure.

mkdir com\microsoft\sqlserver\main

1. Copy module.xml from %TAFJ\_HOME% to the main folder for the corresponding database.

copy %TAFJ\_HOME%\appserver\jboss\jboss7eap\modules\com\microsoft\sqlserver\main\module.xml %JBOSS\_HOME%\modules\com\microsoft\sqlserver\main

1. Copy the corresponding database driver to the main folder of SQL Server.

copy %TAFJ\_HOME%\ext\sqljdbc42.jar %JBOSS\_HOME%\modules\com\microsoft\sqlserver\main

[](http://172.16.204.9/R21UG/Solutions/Runbooks/Stack-06-Runbook-R21/img/8.2.1-01.png)

1. Open module.xml to verify that the driver JAR name matches the folder path for that module.

[](http://172.16.204.9/R21UG/Solutions/Runbooks/Stack-06-Runbook-R21/img/8.2.1-02.png)

### Configuring the TAFJ module

This module contains the TAFJ libraries in lib and ext folders and will have a dependency on the database driver module.

**Procedure**

1. In the File Explorer, in the com folder, create the following directory structure.
2. mkdir temenos\tafj\main
3. Copy the TAFJ module.xml from %TAFJ\_HOME% to the main folder to the %JBOSS\_HOME%\modules folder.

copy %TAFJ\_HOME%\appserver\jboss\jboss7eap\modules\com\temenos\tafj\main\module.xml %JBOSS\_HOME%\modules\com\temenos\tafj\main

1. In %JBOSS\_HOME%\modules\com\temenos\tafj\main, open the command prompt.
2. Create a symbolic link to %TAFJ\_HOME%\lib and %TAFJ\_HOME%\ext.
3. mklink /D lib %TAFJ\_HOME%\lib

mklink /D ext %TAFJ\_HOME%\ext

[](http://172.16.204.9/R21UG/Solutions/Runbooks/Stack-06-Runbook-R21/img/8.2.2-01.png)

1. Open module.xml and uncomment the SQL Server database driver module dependency in the <dependencies> section.

The TAFJ module has dependencies on the Transact module that you need to create. It also contains the MS SQL Server database driver module.

[](http://172.16.204.9/R21UG/Solutions/Runbooks/Stack-06-Runbook-R21/img/8.2.2-02.png)

### Configuring the Transact module

This module contains the Transact JAR files and has a dependency on the TAFJ module created above.

**Procedure**

1. In the File Explorer, in the temenos folder, create this directory structure.
2. mkdir t24\main
3. Open command prompt from %JBOSS\_HOME%\modules\com\temenos\t24\main.
4. Create a symbolic link to %TEMENOS\_HOME%\T24\bnk\t24lib.

mklink /D lib %TEMENOS\_HOME%\T24\bnk\t24lib

[](http://172.16.204.9/R21UG/Solutions/Runbooks/Stack-06-Runbook-R21/img/8.2.3.png)

### Generating the Transact

Unlike the TAFJ module.xml, the Transact module.xml must be generated. You can generate this module using the JBossTools utility that is in %TAFJ\_HOME%\bin.

**Procedure**

1. In %TAFJ\_HOME%\bin, run the following command to create the module.xml file .

JBossTools *module\_name* *path\_to\_jars* *dest* *root\_prefix* -tafjdep

***module\_name***

Name of the JBoss module. Must be the same as the folder path created for the module.

***path\_to\_jars***

The list of folders that you want to parse.

***dest***

The path to the folder where the module.xml will be generated.

***root\_prefix***

Name of the link if module.xml is at the same level as the link.

**-tafjdep**

Option to add TAFJ module dependencies in the dependencies section.

An example is given below.

JBossTools com.temenos.t24 %TEMENOS\_HOME%\T24\bnk\t24lib %JBOSS\_HOME%\modules\com\temenos\t24\main lib –tafjdep

[](http://172.16.204.9/R21UG/Solutions/Runbooks/Stack-06-Runbook-R21/img/8.2.4-01.png)

1. Open module.xml and verify that the TAFJ module is added as a dependency in the <dependencies> section.

[](http://172.16.204.9/R21UG/Solutions/Runbooks/Stack-06-Runbook-R21/img/8.2.4-02.png)

 Dark Mode

[**Expand/Collapse**](http://172.16.204.9/R21UG/Solutions/Runbooks/Stack-06-Runbook-R21/configuring-jboss-eap.htm?tocpath=_____8)

# Configuring a standalone profile on the application server

JBoss Enterprise Application Platform runs in one of two operating modes: as a standalone server or in a managed domain. This runbook covers the standalone installation.

To deploy Transact on TAFJ on EAP, you must configure the various subsystems in Transact.xml. TAFJ uses the management Command Line Interface (CLI) interface tool to automate the process of configuring the J2EE profiles.

**Procedure**

1. In the File Explorer, copy and paste standalone-full.xml into %JBOSS\_HOME%\standalone\configuration and rename it to Transact.xml.
2. Start JBoss using the command given below.
3. %JBOSS\_HOME%\bin\standalone.bat –b 0.0.0.0 –bmanagement 0.0.0.0 --server-config=Transact.xml

**Note:**

As Transact.xml is being used, this command must use the option --server-config to specify the configuration file.

1. Navigate to TAFJ\_HOME%\appserver\jboss\jboss7eap\jboss-cli and open T24Setup.cli in an editor.
2. In the Drivers section, uncomment the driver parameters according to the database used.

[](http://172.16.204.9/R21UG/Solutions/Runbooks/Stack-06-Runbook-R21/img/9-01.png)

1. In the System Property section, comment out the tafj.home property.

[](http://172.16.204.9/R21UG/Solutions/Runbooks/Stack-06-Runbook-R21/img/9-02.png)

1. Open tafj.properties.
2. Enter values for the TAFJ\_HOME, DB\_URL, DB\_DRIVER, DB\_USER, and DB\_PWD parameters.

[](http://172.16.204.9/R21UG/Solutions/Runbooks/Stack-06-Runbook-R21/img/9-03.png)

1. To resolve the user specific system properties in tafj.properties that will be passed as command argument to JBoss CLI command, set the value of <resolve-parameter-values> to true in %JBOSS\_HOME%\bin\jboss-cli.xml. Ensure that <validate-operation-requests> is set to true.

[](http://172.16.204.9/R21UG/Solutions/Runbooks/Stack-06-Runbook-R21/img/9-04.png)

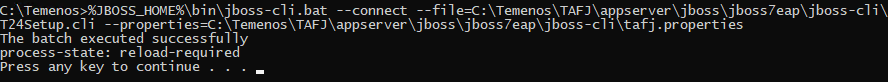
1. To update the Transact.xml file, run the following command:
2. %JBOSS\_HOME%\bin\jboss-cli.bat --connect
3. --file=C:\Temenos\TAFJ\appserver\jboss\jboss7eap\jboss-cli\T24Setup.cli
4. --properties=C:\Temenos\TAFJ\appserver\jboss\jboss7eap\jboss-cli\tafj.properties

You launch JBoss CLI from the %JBOSS\_HOME%\bin folder by running the jboss-cli.bat script.

The complete command takes two arguments, --file and --properties:

* + The --file argument enables CLI commands to be provided from a text file. The %TAFJ\_HOME%\appserver\jboss\jboss7eap\jboss-cli folder has a CLI script file T24Setup.cli. The script has commands to update the profiles specific for Transact deployment.
  + The --properties argument enables passing user-specific properties to the script file from a property file. In the same location you will find tafj.properties.

On successful execution the command updates the required subsystems in Transact.xml (which was used to start up JBoss). If this is successful, you will see text similar to the illustration below.



The scripts now add the following items to the standalone configuration file (Transact.xml):

* + System properties
  + Drivers
  + Data sources
  + Global modules (TAFJ and Transact)
  + JMS Pools
  + JMS Queues

1. Stop the JBoss instance.

The configuration of JBoss with Transact-TAFJ is now complete.

# Extracting UXP Browser artefacts

Before you start installing UXP Browser, you need to prepare its installation packages. This section shows the required steps before deploying Transact and TAFJ artefacts.

**Procedure**

1. Obtain UXP-Browser.202008.zip from your account manager and upload it to the %TEMENOS\_HOME%\UXPBrowser folder on the application server.
2. In this folder, extract UXP-Browser.202008.zip.
3. Locate the IRISR18\_CatalogServices\_TAFJ\_202008.0.1.zip file in your model bank package on your local machine and extract it.
4. Upload the irf-t24catalog-services-202008.0.1.war file to the %UXPBROWSER% directory on your Transact server.
5. Optional: Rename the artefact file names in %UXPBROWSER% by removing the release version information:

| **Before** | **After** |
| --- | --- |
| * + Authenticator-202008.0.1.war   + Browser-202008.0.1.war   + irf-rp-services-202008.0.1.war   + irf-t24catalog-services-202008.0.1.war   + ResourceServerWar-202008.0.1.war | * + Authenticator.war   + Browser.war   + irf-rp-services.war   + irf-t24catalog-services.war   + ResourceServer.war |

**Note:**

You need to rename ResourceServerWar-202008.0.1.war to ResourceServer.war

# Configuring and deploying Transact and TAFJ artefacts

The final step in this setup is to deploy the Transact UI components UXP Browser (UXPB) and BrowserWeb. UXP Browser is the Temenos' latest user interface for working with Transact. BrowserWeb is its forerunner.

**Note:**

This runbook does not cover deployment of other product user interfaces such as Temenos Connect Internet Banking (TCIB) and Mobile Banking (TCMB).

## Deploying TAFJJEE\_EAR.ear and TAFJ Spooler plugins

**Procedure**

1. In the File Explorer, copy TAFJSpoolerPlugins.rar from %TAFJ\_HOME%\appserver to the %JBOSS\_HOME%\standalone\deployments folder.
2. Copy TAFJJEE\_EAR.ear from %TAFJ\_HOME%\appserver\jboss\jboss7eap to the %JBOSS\_HOME%\standalone\deployments folder.

## Deploying BrowserWeb

**Before you begin**

You need to obtain the BrowserWeb-202008.00.tar file from your account manager.

**Procedure**

1. Upload the BrowserWeb-202008.00.tar file from the temporary folder on your machine to the %TEMENOS\_HOME%\Install folder on the application server.
2. Extract the BrowserWeb-202008.00.tar file in the Install folder.
3. Copy the BrowserWeb.war file to the %JBOSS\_HOME%\standalone\deployments folder.

## Modifying Browser.war

**Procedure**

1. In Windows File Explorer, navigate to %UXPBROWSER% and open Browser.war in a compression tool, for example 7-zip.
2. Navigate to WEB-INF and open SSOAPI.properties in an editor.
3. Provide the port number and IP address or host name of your application server where you will deploy Authenticator. Use the actual host IP address and the HTTP port.Example:

SSOAPI.T24REMOTE.URL=http://**10.23.50.XXX:9089**/Authenticator/auth

**Tip:**

As an alternative, you can add the above line in the BRPRuntimeProperties.properties file. This will act as an override and you will not need to modify the Browser.war file.

1. Save the file and close the compression tool.

## Modifying irf-rp-services.war

**Procedure**

1. In Windows File Explorer, navigate to %UXPBROWSER%.
2. Open the irf-rp-services.war file in a compression tool, for example 7-zip and navigate to \WEB-INF\classes.
3. Open connection.properties in your text editor and provide the IP address or host name of your application server, the port, user name and password. Example:
4. # Common EJB Remote Configurations
5. RemoteConnectionHosts=**10.23.50.XXX**
6. RemoteConnectionPorts=**9089**
7. transportLayerUser=**SSOUSER1**

transportLayerCredential=**123456**

1. Save the file and exit the file compression tool.

## Modifying irf-t24catalog-services.war

**Procedure**

1. In Windows File Explorer, navigate to %UXPBROWSER%.
2. Open the irf-t24catalog-services.war file in a compression tool, for example 7-zip, and navigate to \WEB-INF\classes
3. Open connection.properties in your text editor and provide the IP address or host name of your application server, the port, Transact user name and password for that user. Example:
4. # Common EJB Remote Configurations
5. RemoteConnectionHosts=**10.23.50.XXX**
6. RemoteConnectionPorts=**9089**
7. transportLayerUser=**SSOUSER1**

transportLayerCredential=**123456**

1. Save the file and exit the file compression tool.

## Embedding the database driver in ResourceServer.war

**Procedure**

1. In Windows File Explorer, change to %UXPBROWSER%
2. Open the ResourceServer.war file in a compression tool, for example 7-zip, and navigate to \WEB-INF\lib.
3. Copy the database driver file (sqljdbc42.jar) to ResourceServer.war\WEB-INF\lib.
4. Save the file and exit the file compression tool.

## Modifying the required EJB JAR files

**Procedure**

1. To prepare the required EJB JAR files, copy the following three files to the %UXPBROWSER% directory:

copy %TEMENOS\_HOME%\T24\bnk\Extensions\EB\_AuthenticationService\t24ejb\t24-EB\_AuthenticationService-ejb.jar %UXPBROWSER%\

copy %TEMENOS\_HOME%\T24\bnk\Extensions\EB\_CatalogService\t24ejb\t24-EB\_CatalogService-ejb.jar %UXPBROWSER%\

copy %TEMENOS\_HOME%\T24\bnk\Extensions\EB\_ResourceProviderService\t24ejb\t24-EB\_ResourceProviderService-ejb.jar %UXPBROWSER%\

1. Modify t24-EB\_AuthenticationService-ejb.
   1. Open the t24-EB\_AuthenticationService-ejb file in a compression utility and navigate to META-INF.
   2. Open the ejb-jar.xml file in an editor.
   3. In <transaction-type>, specify Bean.
   4. <session-type>Stateless</session-type>
   5. <transaction-type>**Bean**</transaction-type>
   6. In <env-entry-value>, specify either IRISPA or IFPA as the OFS\_SOURCE.
   7. <env-entry>
   8. <description>Comma Separated Runtime Properties for TAFJ Session</description>
   9. <env-entry-name>runtimeProperties</env-entry-name>
   10. <env-entry-type>java.lang.String</env-entry-type>
   11. <env-entry-value>**OFS\_SOURCE=IRISPA**</env-entry-value>

</env-entry>

* 1. Save the file.
  2. Open jboss-ejb3.xml file in an editor and ensure that the security domain is set to other.
  3. <assembly-descriptor>
  4. <s:security>
  5. <ejb-name>AuthenticationServiceBeanTAFJ</ejb-name>
  6. <s:security-domain>**other**</s:security-domain>
  7. </s:security>

</assembly-descriptor>

* 1. Save the file and exit the file compression utility.

1. Modify t24-EB\_CatalogService-ejb.jar.
   1. Open the t24-EB\_CatalogService-ejb.jar file in a compression utility and navigate to META-INF.
   2. Open the ejb-jar.xml file in an editor.
   3. In <transaction-type>, specify Bean.
   4. <session-type>Stateless</session-type>
   5. <transaction-type>**Bean**</transaction-type>
   6. In <env-entry-value>, specify either IRISPA or IFPA as the OFS\_SOURCE.
   7. <env-entry>
   8. <description>Comma Separated Runtime Properties for TAFJ Session</description>
   9. <env-entry-name>runtimeProperties</env-entry-name>
   10. <env-entry-type>java.lang.String</env-entry-type>
   11. <env-entry-value>**OFS\_SOURCE=IRISPA**</env-entry-value>

</env-entry>

* 1. Save the file.
  2. Open jboss-ejb3.xml file in an editor and ensure that the security domain is set to other.
  3. <assembly-descriptor>
  4. <s:security>
  5. <ejb-name>CatalogServiceBeanTAFJ</ejb-name>
  6. <s:security-domain>**other**</s:security-domain>
  7. </s:security>

</assembly-descriptor>

* 1. Save the file and exit the file compression utility.

1. Perform the same steps as above to modify ejb-jar.xml and jboss-ejb3.xml in t24-EB\_ResourceProviderService-ejb.jar.
   1. Open the t24-EB\_ResourceProviderService-ejb.jar file in a compression utility and navigate to META-INF.
   2. Open the ejb-jar.xml file in an editor.
   3. In <transaction-type>, specify Bean.
   4. <session-type>Stateless</session-type>
   5. <transaction-type>**Bean**</transaction-type>
   6. In <env-entry-value>, specify either IRISPA or IFPA as the OFS\_SOURCE.
   7. <env-entry>
   8. <description>Comma Separated Runtime Properties for TAFJ Session</description>
   9. <env-entry-name>runtimeProperties</env-entry-name>
   10. <env-entry-type>java.lang.String</env-entry-type>
   11. <env-entry-value>**OFS\_SOURCE=IRISPA**</env-entry-value>

</env-entry>

* 1. Save the file.
  2. Open jboss-ejb3.xml file in an editor and ensure that the security domain is set to other.
  3. <assembly-descriptor>
  4. <s:security>
  5. <ejb-name>ResourceProviderServiceBeanTAFJ</ejb-name>
  6. <s:security-domain>**other**</s:security-domain>
  7. </s:security>

</assembly-descriptor>

* 1. Save the file and exit the file compression utility.

## Configuring the SSOUSER1 user in JBoss

You need to create the SSOUSER1 user which will be used by the Resource Provider (irf-rp-services.war) and Catalog (irf-t24catalog-services.war) services.

**Procedure**

1. At the Windows command prompt, navigate to %JBOSS\_HOME%\bin.
2. Run the add-user.bat script:

add-user.bat

1. Specify **Application user** (option **b**) as the type of user.
2. C:\Temenos\3rdParty\AS\jboss-eap-7.3\bin>**add-user.bat**
3. What type of user do you wish to add?
4. a) Management User (mgmt-users.properties)
5. b) Application User (application-users.properties)

**(a): b**

1. Provide the user name (SSOUSER1) and password (123456).
2. Enter the details of the new user to add.
3. Using realm 'ApplicationRealm' as discovered from the existing property files.
4. Username : SSOUSER1
5. Password recommendations are listed below. To modify these restrictions edit the add-user.properties configuration file.
6. - The password should be different from the username
7. - The password should not be one of the following restricted values {root, admin, administrator}
8. - The password should contain at least 8 characters, 1 alphabetic character(s), 1 digit(s), 1 non-alphanumeric symbol(s)

Password :

1. Confirm that you want to use the password and re-enter it.
2. WFLYDM0099: Password should have at least 8 characters!
3. Are you sure you want to use the password entered yes/no? **yes**

Re-enter Password :

1. Specify the group the user should belong to; here t24user.
2. What groups do you want this user to belong to? (Please enter a comma separated list, or leave blank for none)[ ]: **t24user**
3. Confirm the addition of the user to the realm ApplicationRealm by providing the answer yes.
4. About to add user 'SSOUSER1' for realm 'ApplicationRealm'

Is this correct yes/no? **yes**

The user has been added to JBoss configuration files.

Added user 'SSOUSER1' to file 'C:\Temenos\3rdParty\AS\jboss-eap-7.3\standalone\configuration\application-users.properties'

Added user 'SSOUSER1' to file 'C:\Temenos\3rdParty\AS\jboss-eap-7.3\domain\configuration\application-users.properties'

Added user 'SSOUSER1' with groups t24user to file 'C:\Temenos\3rdParty\AS\jboss-eap-7.3\standalone\configuration\application-roles.properties'

Added user 'SSOUSER1' with groups t24user to file 'C:\Temenos\3rdParty\AS\jboss-eap-7.3\domain\configuration\application-roles.properties'

1. For the last question, provide the answer yes.
2. Is this new user going to be used for one AS process to connect to another AS process?
3. e.g. for a slave host controller connecting to the master or for a Remoting connection for server to server EJB calls.

yes/no? **yes**

The user has been created and the following message is displayed at the command prompt:

To represent the user add the following to the server-identities definition **<secret value="MTIzNDU2" />**

1. Open the Transact.xml file that is in %JBOSS\_HOME%\standalone\configuration in a text editor and add under <server-identities> the tag that you generated in the previous step:
2. <server-identities>
3. <ssl>
4. <keystore path="application.keystore" relative-to="jboss.server.config.dir" keystore-password="password" alias="server" key-password="password" generate-self-signed-certificate-host="localhost"/>
5. </ssl>
6. **<secret value="MTIzNDU2" />**
7. </server-identities>

## Configuring UXP Browser in the Generate On Demand (GOD) mode

**Before you begin**

You need to locate the encryption utility (EncryptPassword.jar) in your Model Bank package and then upload it to the $UXPBROWSER directory on the application server. The JAR file is available at the following location:  
*Model\_Bank\_package*.zip\UXP-Browser.zip\UXPB-Tools.zip\EncryptPassword.jar

**Procedure**

1. At the command line, navigate to %TEMENOS\_HOME% and create the properties directory.
2. Go to the %UXPBROWSER% directory and extract the uxpb-god-properties-202008.0.1.zip file.
3. Copy the two properties files to the properties directory where they will be stored permanently:
4. Encrypt the IRIS system and JDBC user passwords using the Temenos encryption utility EncryptPassword.jar – run the following command:

java -jar EncryptPassword.jar *pwd\_to\_be\_encrypted*

1. Save the encrypted password in a file in a secure location.
2. Edit the BRPRuntimeProperties.properties file in a text editor - perform the following steps:
   1. Provide the Transact user name and the encrypted password (here 123456):
   2. browser.options.irisSystemUser=**INPUTT**

browser.options.irisSystemPassword=**vSd3/5UdSJc=**

* 1. Uncomment the following lines and provide the Resource Provider, Catalog services, Resource Server and Authenticator URLs with the IP address or host name of your application server together with the port number:
  2. browser.options.dynamicIRIS.rpService=**http://10.23.50.XXX:9089/irf-rp-services**
  3. browser.options.god.catalogService=**http://10.23.50.XXX:9089/irf-t24catalog-services**
  4. browser.options.resourceServer.url=**http://10.23.50.XXX:9089/ResourceServer**

browser.options.remoteAuth.url=**http://10.23.50.XXX:9089/Authenticator/auth/logon**

* 1. Save the file.

1. Edit the RSDefaultProperties.properties file in a text editor – provide the following and save the file:
   1. Database URL
   2. Database user name
   3. The encrypted password for the database user
   4. JDBC driver name for MS SQL Server
   5. Blob type
   6. Long type

Example:

resource.server.options.tenant.jdbc.url.1=**jdbc:sqlserver://10.23.50.XXX:1433;databaseName=MB202008;integratedSecurity=false**

resource.server.options.tenant.jdbc.username.1=**tafj**

resource.server.options.tenant.jdbc.password.1=**DySX79VV3rQ=**

resource.server.options.tenant.jdbc.driver.1=**com.microsoft.sqlserver.jdbc.SQLServerDriver**

resource.server.options.tenant.ddl.blobType.1=**varbinary(max)**

resource.server.options.tenant.ddl.longType.1=**bigint**

## Deploying UXP Browser artefacts in JBoss

Browser.war is the main UXP Browser artefact. UXP Browser requests are routed through Interaction Framework, so along with the Browser, you must also deploy the Authenticator application , the Resource Provider and Interaction Framework catalog services and also Resource server. This section is not applicable if you have skipped [**Extracting UXP Browser artefacts**](http://172.16.204.9/R21UG/Solutions/Runbooks/Stack-06-Runbook-R21/extracing-uxp-browser-artefacts.htm).

**Procedure**

1. At the Windows command prompt, change to %UXPBROWSER%.
2. Deploy the Authenticator component.

copy Authenticator.war %JBOSS\_HOME%\standalone\deployments

1. Deploy Browser.war.

copy Browser.war %JBOSS\_HOME%\standalone\deployments

1. Deploy Interaction Framework Resource Provider services.

copy irf-rp-services.war %JBOSS\_HOME%\standalone\deployments

1. Deploy Interaction Framework Catalog services.

copy irf-t24catalog-services.war %JBOSS\_HOME%\standalone\deployments

1. Deploy Resource Server.

copy ResourceServer.war %JBOSS\_HOME%\standalone\deployments

## Deploying the EJB JAR files in JBoss

**Procedure**

1. Deploy the Authentication service.

cd %UXPBROWSER%

copy t24-EB\_AuthenticationService-ejb.jar %JBOSS\_HOME%\standalone\deployments

1. Deploy the Catalog Service component.

copy t24-EB\_CatalogService-ejb.jar %JBOSS\_HOME%\standalone\deployments

1. Deploy the Resource Provider service.

copy t24-EB\_ResourceProviderService-ejb.jar %JBOSS\_HOME%\standalone\deployments

# Deploying Transact component services

You will find the Transact component service APIs as Axis2 archives (t24-*component\_service*-jws.aar) in the extensions folder of the Transact distribution pack you have received. Deploy the archives into the Apache Axis2 web archive file. The release version of Apache Axis2 used in this runbook is 1.6.2. The topics in this chapter show you how to package and deploy axis2.war on your JBoss application server.

## Step 1: Packaging Axis2 archives

**Procedure**

1. From the Apache website download axis2-1.6.2-war.zip.
2. Extract the zip file.
3. In File Explorer, copy axis2.war to any directory (e.g. UXPBrowser).
4. Open axis2.war in a file compression tool, for example 7-zip, and navigate to WEB-INF\lib.
5. Add the following dependency JAR files to the lib folder.
   * commons-pool-1.5.5.jar
   * spring-2.5.6.jar

[](http://172.16.204.9/R21UG/Solutions/Runbooks/Stack-06-Runbook-R21/img/12.1-00.png)

1. Copy the following component service archives from the subdirectories under %TEMENOS\_HOME%\T24\bnk\Extensions into the WEB-INF\services directory of axis2.war.
   * t24-DS\_DesignStudioInstallerService-tafj-jws.aar
   * t24-EB\_CatalogService-tafj-jws.aar
   * t24-EB\_OFSConnectorService-tafj-jws.aar
   * t24-EB\_ResourceProviderService-tafj-jws.aar
   * t24-IF\_IntegrationFlowService-tafj-jws.aar
   * t24-IF\_IntegrationFrameworkService-tafj-jws.aar
   * t24-IF\_IntegrationLandscapeService-tafj-jws.aar
2. Open WEB-INF\services\services.list and add the names of the component service archives that you added to the services directory.

A screenshot of a browser

Description automatically generated

1. Open WEB-INF\web.xml and configure database connectivity by adding the data source.
2. <!-- Temenos T24 Service Provider Specific Settings -->
3. <resource-ref id="**ResourceRef\_t24DataSource**">
4. <description>Used to get connections from T24 jdbc pool</description>
5. <res-ref-name>**jdbc/t24DataSource**</res-ref-name>
6. <res-type>**javax.sql.DataSource**</res-type>
7. <res-auth>**Container**</res-auth>
8. </resource-ref>
9. <resource-ref id="**ResourceRef\_t24LockingDataSource**">
10. <description>Used to get connections from T24 jdbc pool</description>
11. <res-ref-name>**jdbc/t24LockingDataSource**</res-ref-name>
12. <res-type>**javax.sql.DataSource**</res-type>
13. <res-auth>**Container**</res-auth>

</resource-ref>

1. In WEB-INF add the the JBoss-specific deployment descriptor ( ).
2. Configure database connectivity by adding the data sources.
3. <?xml version="1.0" encoding="UTF-8"?>
4. <jboss-web>
5. <resource-ref>
6. <res-ref-name>**jdbc/t24DataSource**</res-ref-name>
7. <res-type>**javax.sql.DataSource**</res-type>
8. <jndi-name>**java:/jdbc/t24Axis2DS**</jndi-name>
9. </resource-ref>
10. <resource-ref>
11. <res-ref-name>**jdbc/t24LockingDataSource**</res-ref-name>
12. <res-type>**javax .sql.DataSource**</res-type>
13. <jndi-name>**java:/jdbc/t24LockingDS**</jndi-name>
14. </resource-ref>

<jboss-web>

1. Save the changes to axis2.war.

## Step 2: Deploying Axis2

**Procedure**

1. In the Windows File Explorer, navigate to %JBOSS\_HOME%\standalone\configuration.
2. Open Transact.xml.
3. In datasource, add the data source for axis2.war deployment.
4. <datasource jta="true" judo-name="**java:/jdbc/t24Axis2DS**" pool-name="**Axis2DS**" enabled="true" use-java-context="true" use-ccm="true">
5. <connection-url>**jdbc:sqlserver://10.23.50.XXX:1433;databaseName=MBR20;integratedSecurity=false**</connection-url>
6. <connection-property name="defaultRowPrefetch">**100**</connection-property>
7. <driver>**sqljdbc**</driver>
8. <pool>
9. <min-pool-size>**5**</min-pool-size>
10. <max-pool-size>**220**</max-pool-size>
11. <flush-strategy>**FailingConnectionOnly**</flush-strategy>
12. </pool>
13. <security>
14. <user-name>**tafj**</user-name>
15. <password>**xxxx**</password>
16. </security>

</datasource>

1. Copy axis2.war from the UXPBrowser folder to %JBOSS\_HOME%\standalone\deployments.

copy axis2.war %JBOSS\_HOME%\standalone\deployments

**Result**: You have now deployed all the required components.

[](http://172.16.204.9/R21UG/Solutions/Runbooks/Stack-06-Runbook-R21/img/12.1-03.png)

# Sanity check - post deployment

Log in to Transact through the product user interfaces to verify your application server deployment.  This section shows the steps to login using UXP Browser and BrowserWeb.

## Starting the application server

**Procedure**

1. In %JBOSS\_HOME%\bin, run standalone.bat to start JBoss.

If you are not deploying UXP Browser, the following are sufficient.

%JBOSS\_HOME%\bin\standalone.bat --server-config=Transact.xml –b 0.0.0.0 -Djboss.http.port=9089

If you are deploying UXP Browser, the following are required. (Each command argument has been placed in a new line for brevity.)

%JBOSS\_HOME%/bin/standalone.bat

--server-config=Transact.xml

-Djboss.http.port=9089

-Djboss.node.name=node1

-b 0.0.0.0

-DedgeSystemPropertyFolder=%TEMENOS\_HOME%\properties

-DedgeSystemPropertyUpdaters=

com.temenos.connect.system.GODSystemTestProperties

-DBRP\_HOME=%TEMENOS\_HOME%

**Note:**

Optionally you may set JBoss HTTP port (-Djboss.http.port=9089) to start JBoss at a user-defined port instead of the default HTTP port 8080.

1. When the server is started, in the %JBOSS\_HOME%\standalone\deployments folder check that all the artefacts listed have been deployed.

### Mandatory command arguments

**--server-config**

Indicates standalone profile configuration file (for example Transact.xml) to use at JBoss startup.

**-Djboss.http.port**

Port number of the application server.

**-Djboss.node.name**

Unique identifier for the application server instance.

**-DedgeSystemPropertyUpdaters**

This argument is used to set the required properties to be loaded. For GOD mode, use the value below:

com.temenos.connect.system.GODSystemTestProperties

## Accessing the TAFJEE servlet

The TAFJJEE\_EAR.ear file contains a WAR file, which is a helper servlet for configuration, diagnostic, execution and troubleshooting related TAFJ functions on the application server. TAFJEE has protected access and is subject to BASIC authentication.

**Procedure**

1. Open the command prompt in %JBOSS\_HOME%\bin.
2. Run add-user.bat to create an application user.
3. Give the user **TAFJAdmin** role.
   1. Specify the type of user, here **Application** user.
   2. Provide the preferred user name and password.
   3. Specify the groups the user should belong to; here TAFJAdmin.
   4. Confirm the addition of the user to the realm ApplicationRealm by providing the answer yes.
   5. For the last question, provide the answer no.
4. Use URL http://*host\_name\_or\_IP\_address*:9089/TAFJEE to access the TAFJEE servlet.
5. Enter the application user credentials.

[](http://172.16.204.9/R21UG/Solutions/Runbooks/Stack-06-Runbook-R21/img/13.2-01.png)

This gives you access to all the functions in TAFJEE.

[](http://172.16.204.9/R21UG/Solutions/Runbooks/Stack-06-Runbook-R21/img/13.2-02.png)

**Note:**

Some of the tools are protected and require TAFJ user credentials.

1. Click **tShow** to view the compilation details of a Transact routine on the application server.

[](http://172.16.204.9/R21UG/Solutions/Runbooks/Stack-06-Runbook-R21/img/13.2-03.png)

## Accessing BrowserWeb

**Procedure**

1. Access BrowserWeb using the following URL:

http://*host\_name\_or\_IP\_address*:9089/BrowserWeb

.

1. Log in to BrowserWeb using a valid Transact user name and password.

[](http://172.16.204.9/R21UG/Solutions/Runbooks/Stack-06-Runbook-R21/img/13.3-01.png)

The landing page is displayed.

[](http://172.16.204.9/R21UG/Solutions/Runbooks/Stack-06-Runbook-R21/img/13.3-02.png)

1. Launch a Transact application, for example SPF S SYSTEM.

[](http://172.16.204.9/R21UG/Solutions/Runbooks/Stack-06-Runbook-R21/img/13.3-03.png)

## Accessing UXP Browser

**Procedure**

1. Use the following URL to access UXP Browser.

http://*host\_name\_or\_IP\_address*:9089/Browser

[](http://172.16.204.9/R21UG/Solutions/Runbooks/Stack-06-Runbook-R21/img/13.4-01.png)

1. Log in to Browser using a valid Transact user name and password. The landing page is now displayed.

[](http://172.16.204.9/R21UG/Solutions/Runbooks/Stack-06-Runbook-R21/img/13.4-02.png)

1. Launch a Transact application, for example SPF S SYSTEM.

[](http://172.16.204.9/R21UG/Solutions/Runbooks/Stack-06-Runbook-R21/img/13.4-03.png)

## Using the Axis2 servlet

### Accessing the Axis2 servlet

**Procedure**

1. Use the following URL to access the Axis2 servlet.

http://*host\_name\_or\_IP\_address*:9089/axis2

[](http://172.16.204.9/R21UG/Solutions/Runbooks/Stack-06-Runbook-R21/img/13.5.1-01.png)

1. Click **Validate**. A new page opens.

[](http://172.16.204.9/R21UG/Solutions/Runbooks/Stack-06-Runbook-R21/img/13.5.1-02.png)

1. Check whether the web services are working correctly.
2. Go back to the main page and click **Services**. The services deployed in Axis2 are displayed.

[](http://172.16.204.9/R21UG/Solutions/Runbooks/Stack-06-Runbook-R21/img/13.5.1-03.png)

### Testing Axis2 and Transact

**Procedure**

1. In the Axis2 servlet, select **Services**.
2. Select a service to display the WSDL file (the example illustrated below is for IntegrationFlowService).

[](http://172.16.204.9/R21UG/Solutions/Runbooks/Stack-06-Runbook-R21/img/13.5.2-01.png)

1. Using a third-party tool, such as SOAPUI, create a new SOAP project using WSDL.

[](http://172.16.204.9/R21UG/Solutions/Runbooks/Stack-06-Runbook-R21/img/13.5.2-02.png)

1. Create a new read flow.

[](http://172.16.204.9/R21UG/Solutions/Runbooks/Stack-06-Runbook-R21/img/13.5.2-04.png)

This completes the deployment of Transact on TAFJ on JBoss EAP 7.3 with the SQL Server 2019 database. You will now be able to perform business operations in Transact using BrowserWeb and UXP Browser.