TEMENOS

THE BANKING SOFTWARE COMPANY

Stack 6 for JBoss - Windows

Customer Runbook 2.0 December 2019

All post R19 AMR releases up to and including R20 AMR

Contents

1 About this Runbook

The **Stack 6 for JBoss - Windows Runbook** describes how to configure TAFJ-Transact Transact on a JBoss-EAP-7.2 with SQL Server 2017 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 R19 AMR releases up to and including R20 AMR.

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

1.1 Stacks table

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

1.2 Scope

This runbook covers:

- Installing TAFJ.
- Installing Transact.
- Configuring JBoss EAP 7.2.
- Deploying Transact and TAFJ artefacts in JBoss.

1.3 Audience

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

1.3.1 Skills and knowledge

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

- Transact
- TAFJ
- JBoss
- MSSQL Server

1.3.2 Other stack runbooks

You can find all our stack runbooks on the Stack verification home page at http://uni-t.temenos.com/product/Stack-Verification.

1.4 Legal

The information in this guide relates to TEMENOS™ information, products and services. It also includes information, data and keys developed by other parties.

While all reasonable attempts have been made to ensure accuracy, currency and reliability of the content in this guide, all information is provided "as is".

There is no guarantee as to the completeness, accuracy, timeliness or the results obtained from the use of this information. No warranty of any kind is given, expressed or implied, including, but not limited to warranties of performance, merchantability and fitness for a particular purpose.

In no event will TEMENOS be liable to you or anyone else for any decision made or action taken in reliance on the information in this document or for any consequential, special or similar damages, even if advised of the possibility of such damages.

TEMENOS does not accept any responsibility for any errors or omissions, or for the results obtained from the use of this information. Information obtained from this guide should not be used as a substitute for consultation with TEMENOS.

References and links to external sites and documentation are provided as a service. TEMENOS is not endorsing any provider of products or services by facilitating access to these sites or documentation from this guide.

The content of this guide is protected by copyright and trademark law. Apart from fair dealing for the purposes of private study, research, criticism or review, as permitted under copyright law, no part may be reproduced or reused for any commercial purposes whatsoever without the prior written permission of the copyright owner. All trademarks, logos and other marks shown in this guide are the property of their respective owners.

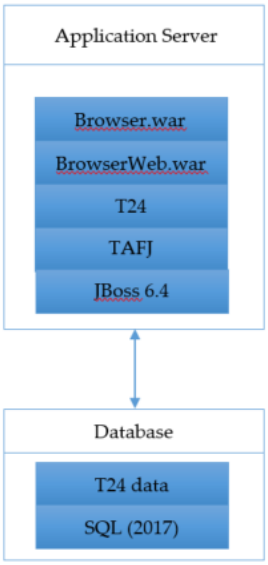
1.5 History

Version	Date	Description	Author
1.0	July 2019	Initial release of this runbook.	Radhika Bonthu and Robert Norledge
2.0	December 2019	Added "Additional modifications for release 201910 onwards" on page 59.	Suvalakshmi V

2 Introduction

This runbook helps you to set up the Dev/Test Environment for Transact/TAFJ on the Windows operating system. It guides you through the installation and configuration of TAFJ, Transact and JBoss to access the Transact browser. In this runbook, a 2-tier architecture is followed with an Application server and a Database server.

The Application tier contains the JBoss EAP (version 7.2) application server with TAFJ/Transact. The Database tier contains the SQL (2017) database that stores Transact data. For better understanding the architecture is given below.



NOTE:
If you are interested in setting up a Pre-Prod/Prod system then please refer to the reference architecture documentation on the Stack verification site.

3 Prerequisites and assumptions

3.1 Prerequisites

Before you start to install and configure the stack, you must have installed all 3rd Party software and copied all Temenos artefacts from Distribution.

3.1.1 Third party software

Software	Version
Windows server	2016
JDK	1.8
JBoss	7.2
SQL server	2017
SQL server management studio	2017

3.1.2 Temenos artefacts

Artefacts	File	Description
201905.bak	`MSSQL201905.bak`	Contains the database.
Transact	`MB.201905.TAFJ201905.Internal.bnk.tar`	Contains the bnk directory that holds the Transact libraries.
TAFJ	`TAFJ.DEV.201905.0.tar.gz`	Contains the TAFJ runtime .jar file, TAFJ patch script and TAFJ setup script.
Transact Browser	`BrowserWeb-201905.00.tar`	Old browser components.
UXPBrowser	`UXP-Browser.201905.zip`	New browser components.
Axis2	`axis2.war`	Contains Web services.

3.2 Assumptions

This runbook assumes that:

- The Transact data has been loaded into the SQL server from the `.bak` file.
- The database has been loaded with TAFJ related stored procedures/java functions by following the database specific installation document in %TAFJ_HOME%/doc.
- All 3rd party software is installed. See "Creating the directory structure" on page 12 for the installation path.
- All Temenos artefacts are copied to the desired installation path. See "Creating the directory structure" on page 12 for the installation path.

4 Before you start

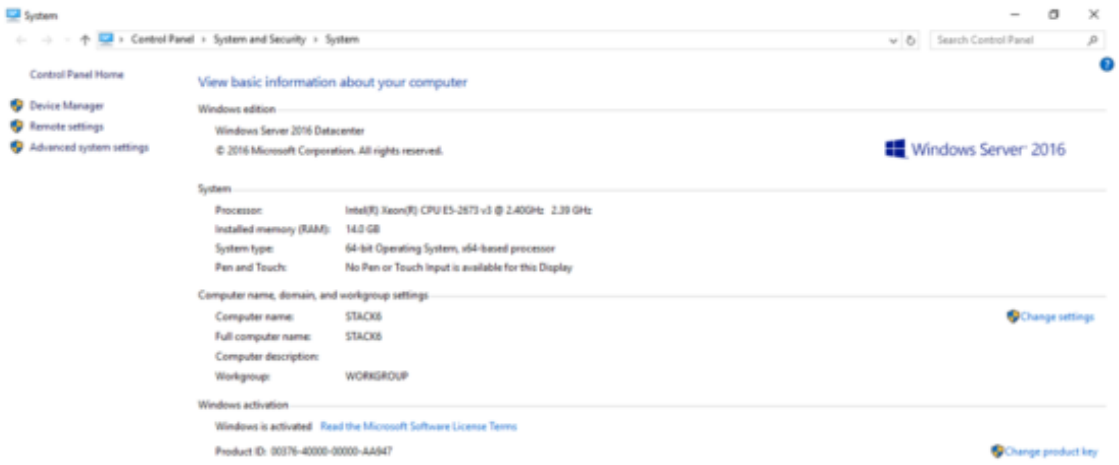
Before you start installing and configuring the stack, you need to check that your technical infrastructure has been set up properly.

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

4.1 Checking the operating system

Procedure

1. Display the Windows Control Panel.
2. Select **System and Security**.
3. Click **System**.



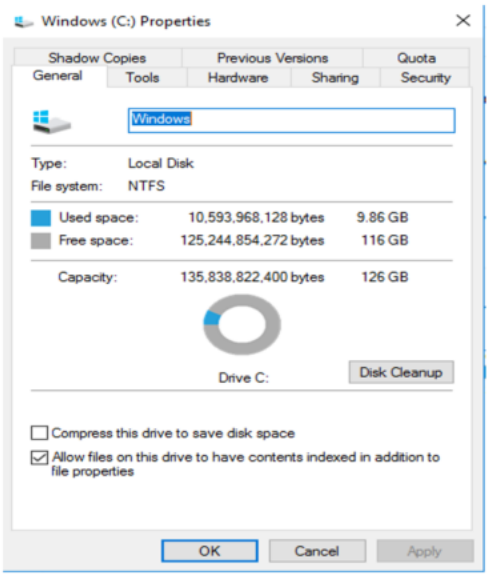
4. Check that the operating system is Windows Server 2016.

4.2 Checking the disk space

Procedure

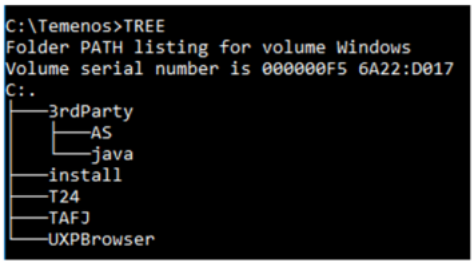
1. In the File Explorer, right-click the C drive.
2. Click **Properties**.

   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 20GB.



4.3 Creating the directory structure

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



The following table explains the directory structure:

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

4.4 Configuring the environment variables

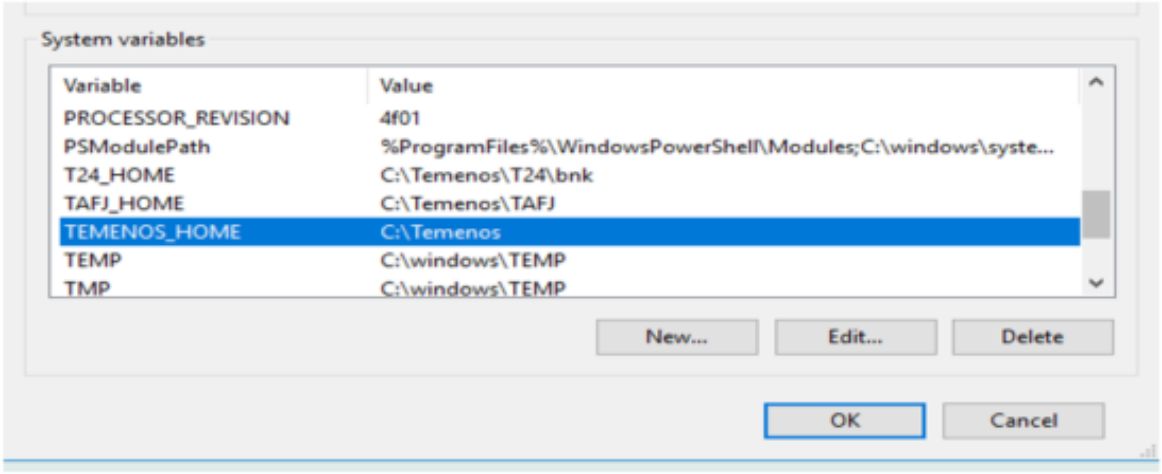
Procedure

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.

- **TEMENOS_HOME=C:\Temenos**
- **T24_HOME=%TEMENOS_HOME%\T24\bnk**
- **TAFJ_HOME=%TEMENOS_HOME%\TAFJ**
- **UXPBROWSER=%TEMENOS_HOME%\UXPBrowser**



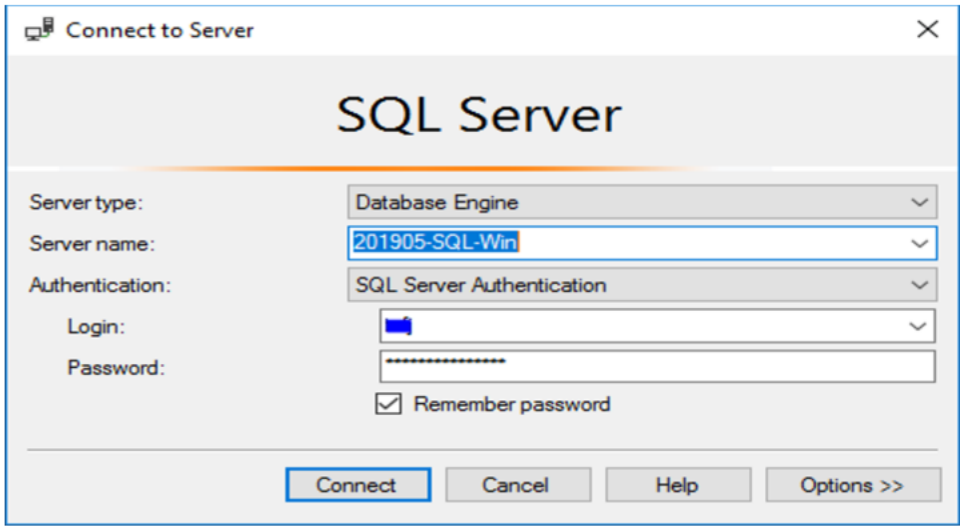
4.5 Verifying the system for third party software

Procedure

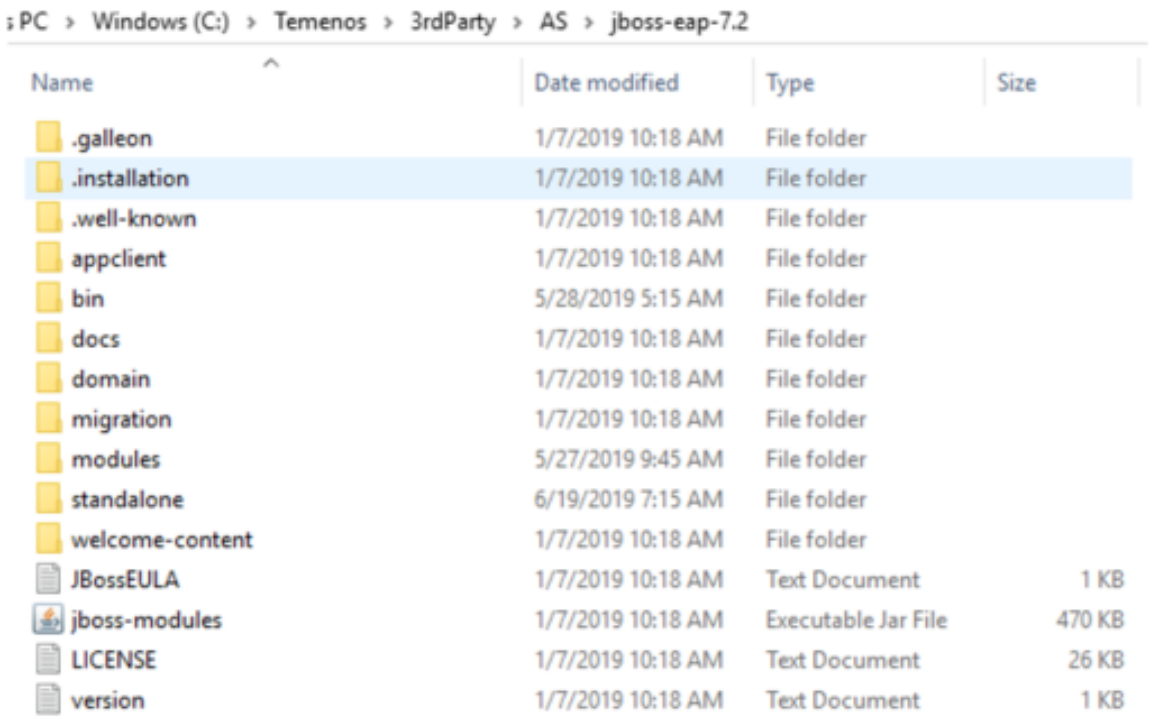
1. To verify that JDK 1.8 is installed, at the Command prompt enter `java -version`. Add the JAVA_HOME variable and `%JAVA_HOME%\bin` to the `%PATH%` variable. You should see the Java version displayed.

```
C:\Temenos>java -version
java version "1.8.0_211"
Java(TM) SE Runtime Environment (build 1.8.0_211-b12)
Java HotSpot(TM) 64-Bit Server VM (build 25.211-b12, mixed mode)
```

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



3. To verify that JBoss EAP 7.2 is installed, and that JBOSS_HOME has been added as an environment variable, check that the following folders are present in *%JBOSS_HOME%*.

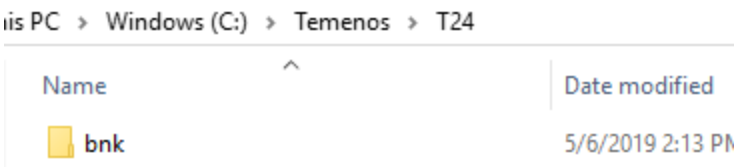


5 Installing TAFJ runtime

5.1 Extracting Transact

Procedure

1. Copy the Transact artefacts (`MB.201905.TAFJ201905.Internal.bnk.tar.gz`) from distribution to folder `%TEMENOS_HOME%\install`.
2. Open `%TEMENOS_HOME%\install`.
3. Extract the tar file to the `T24` directory (`C:\Temenos\T24`). The file is extracted as a `bnk` folder.

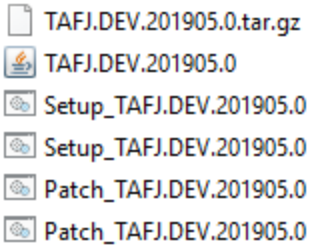


5.2 Installing TAFJ

> **NOTE:** The only pre-requisite for TAFJ installation is to have JAVA_HOME set to the correct path.

Procedure

1. Copy the TAFJ artefacts (`TAFJ.DEV.201905.0.tar.gz`) from Distribution to folder `%TEMENOS_HOME%\install`.
2. In *`%TEMENOS_HOME%`*`\install` extract `TAFJ.DEV.201905.0.tar.gz`.



3. In the *%TEMENOS_HOME%*\install folder, open the command prompt.

4. Execute Setup_TAFJ.DEV.201905.0.bat.

5. During the installation, when prompted for the installation directory, enter the path in *%TAFJ_HOME%*.

```
Microsoft Windows [Version 10.0.14393]
(c) 2016 Microsoft Corporation. All rights reserved.

C:\Temenos\install>Setup_TAFJ.DEV.201905.0.bat
------------------
YAJI v R18
WARNING : JDK 1.8 or above required
Starting setup...
-------------------------------------------------------------
.    Welcome to the Installation program of TAFJ            .
. This setup will install the necessary libraries to        .
. compile and run BASIC programs as well as some            .
. additional tools and Eclipse plug-ins.                    .
. Note that nothing will be modified outside of the         .
. directory you will specify for the installation.          .
. To fully uninstall it, just delete the directory.         .
-------------------------------------------------------------

Press ENTER to continue.

-------------------------------------------------------------
Please enter the install Directory.
 default : 'C:\Temenos\install/TAFJ'
C:\Temenos\TAFJ
```

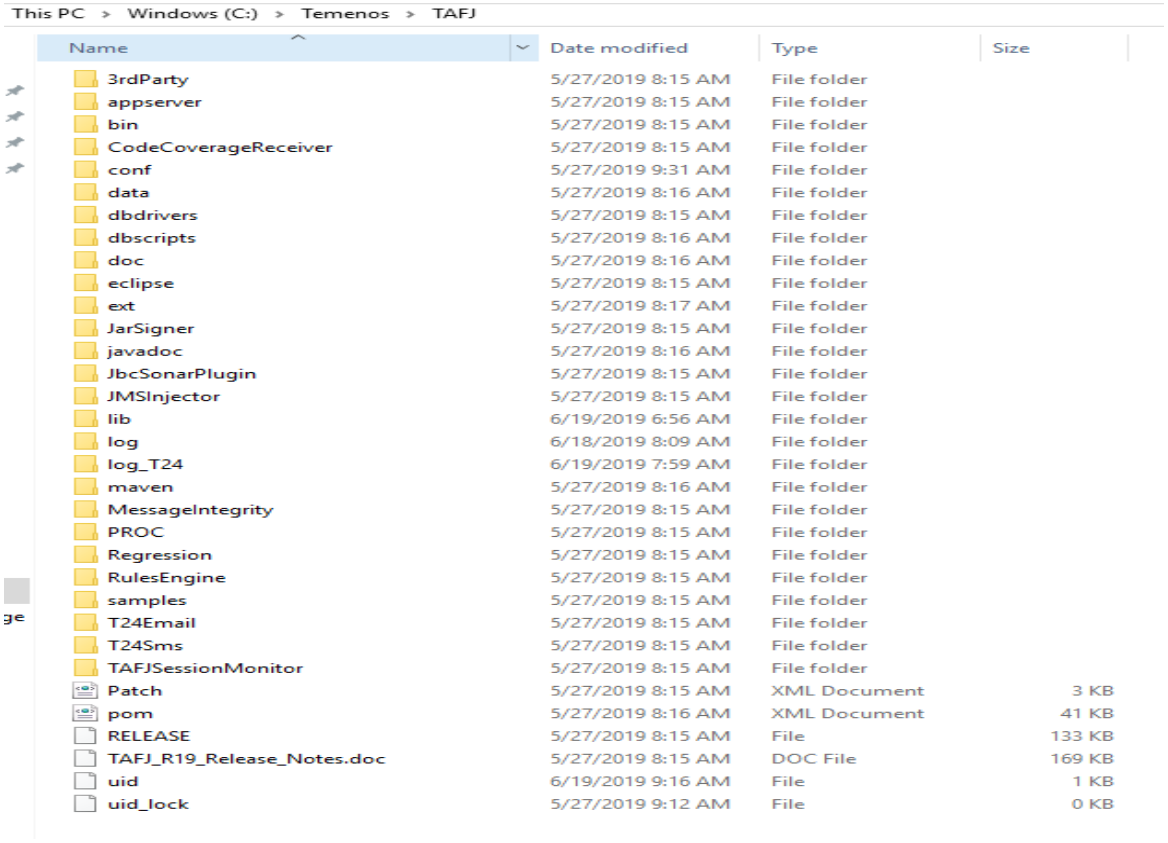
6. Press Enter at the next prompt for eclipse home directory. Enter y to create a conf directory.

```
------------------------------------------------------------
Please enter the eclipse home directory (Optional).
When provided, a tafj.link, used to resolve tafj eclipse
plugins will be created under ECLIPSE_HOME/dropins
If this file already exist it will be replaced
 default : 'C:\Temenos\TAFJ/conf'


'C:\Temenos\TAFJ\conf' doesn't exist. Create (y/n) ?
y
```

7. Press Enter to terminate. This completes the TAFJ runtime installation.

```
Getting available devices ....   [DONE]
Creating first project "tafj" ...   [DONE]

Updating : 'C:\Temenos\TAFJ\conf\tafj.properties'.
Updating : 'C:\Temenos\TAFJ\conf\.properties'.
Updating : 'C:\Temenos\TAFJ\dbscripts\oracle\javaload'.
Updating : 'C:\Temenos\TAFJ\dbscripts\oracle\javaload.bat'.
Updating : 'C:\Temenos\TAFJ\dbscripts\oracle\javaunload'.
Updating : 'C:\Temenos\TAFJ\dbscripts\oracle\javaunload.bat'.
Updating : 'C:\Temenos\TAFJ\dbscripts\oracle\javaloadSingle'.
Updating : 'C:\Temenos\TAFJ\dbscripts\oracle\javaloadSingle.bat'.
Updating : 'C:\Temenos\TAFJ\dbscripts\oracle\javaunloadSingle'.
Updating : 'C:\Temenos\TAFJ\dbscripts\oracle\javaunloadSingle.bat'.
ECLIPSE_HOME: C:\Temenos\TAFJ/conf
tafj.link created: C:\Temenos\TAFJ/conf
============================================================
.          Thank you for having installed TAFJ             .
. *** IMPORTANT ***                                        .
. Please make sure that you have the environment variable .
. JAVA_HOME set to a JDK 8 or above.                       .
============================================================

Press ENTER to terminate.
```

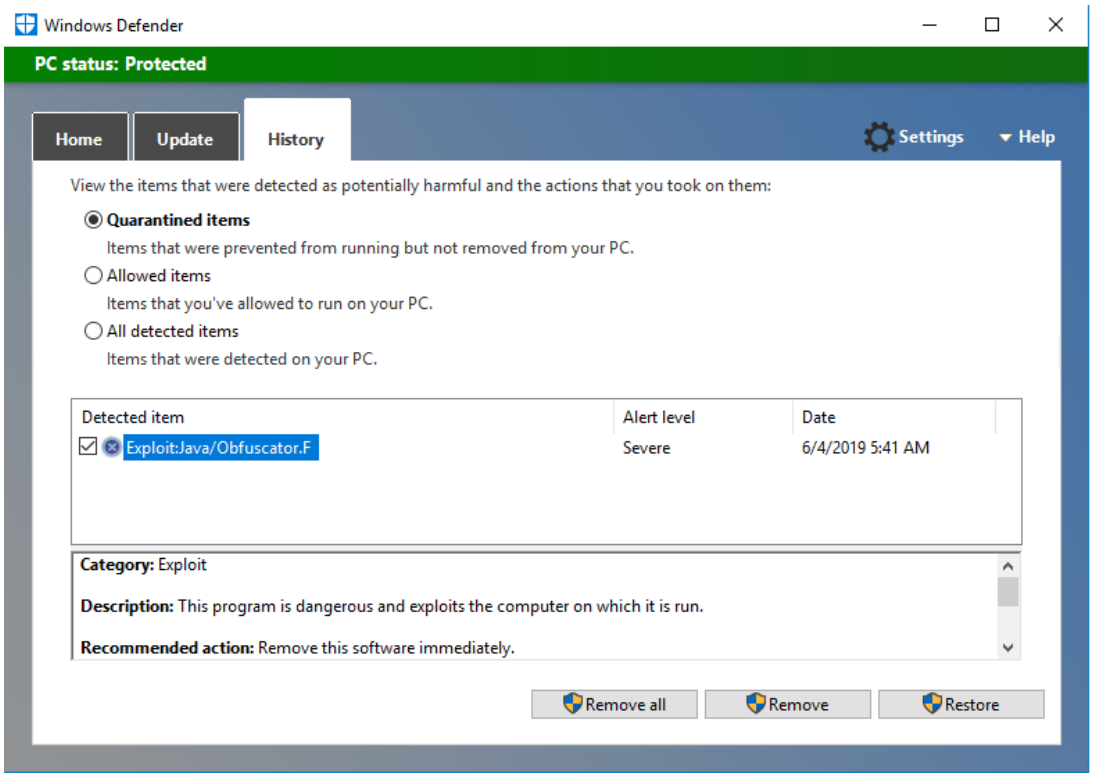
You should see the following folders in *%TAFJ_HOME%*.



8. In the File Explorer, open the command prompt from `%TAFJ_HOME%\bin`. Execute the tVersion command to verify the TAFJ version.

```
<ivy-module version="1.0">
<info organisation="Temenos" module="TAFJ"  branch="R19_ALL"/>
        <dependencies>
                <dependency org="Temenos" branch="DEV_201905" name="FJ_TAFJCache" rev="0" />
                <dependency org="Temenos" branch="DEV_201905" name="FJ_TAFJClient" rev="0" />
                <dependency org="Temenos" branch="DEV_201905" name="FJ_TAFJCommon" rev="0" />
                <dependency org="Temenos" branch="DEV_201905" name="FJ_TAFJCompiler" rev="0" />
                <dependency org="Temenos" branch="DEV_201905" name="FJ_TAFJConfiguration" rev="0" />
                <dependency org="Temenos" branch="DEV_201905" name="FJ_TAFJCore" rev="0" />
                <dependency org="Temenos" branch="DEV_201905" name="FJ_TAFJDBImport" rev="0" />
                <dependency org="Temenos" branch="DEV_201905" name="FJ_TAFJHealthCheck" rev="0" />
                <dependency org="Temenos" branch="DEV_201905" name="FJ_TAFJLocking" rev="0" />
                <dependency org="Temenos" branch="DEV_201905" name="FJ_TAFJLogging" rev="0" />
                <dependency org="Temenos" branch="DEV_201905" name="FJ_TAFJPackager" rev="0" />
                <dependency org="Temenos" branch="DEV_201905" name="FJ_TAFJVersion" rev="0" />
                <dependency org="Temenos" branch="DEV_201905" name="FJ_TAFJBASIC" rev="0" />
                <dependency org="Temenos" branch="DEV_201905" name="FJ_tComponentFramework" rev="0" />
        </dependencies>
</ivy-module>C:\Temenos\TAFJ\bin>
```

9. In Windows Defender restore the quarantined TemenosSecurity.jar as shown below:



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 `Malware Detected` message.

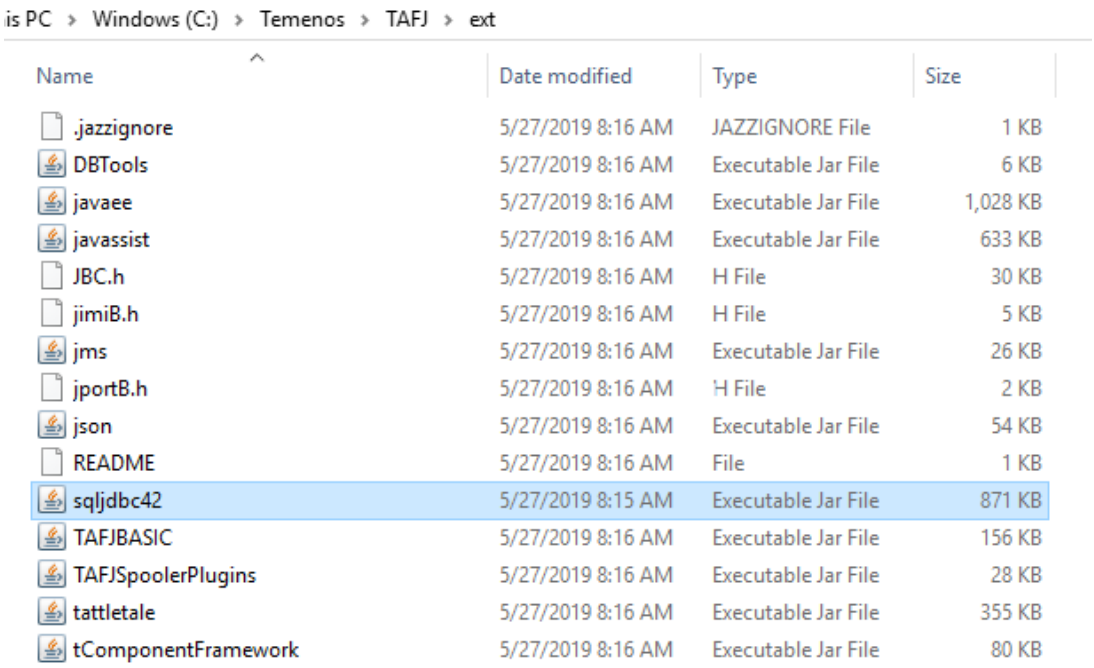
5.3 Deploying the database driver

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 vendor's portal.

Procedure

1. Download the JDBC sql driver from Microsoft portal.
2. Copy `sqljdbc42.jar` to `%TAFJ_HOME%\ext folder`.



6 Configuring TAFJ

This chapter covers how to configure TAFJ runtime to setup 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 *`%TAFJ_HOME%`*`/conf` directory. The default property file is `tafj.properties`.

6.1 Configuring tafj.properties

> **NOTE:** For more information on `tafj.properties` please refer to *`%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.

```
# Directories
#
#*****************************************************
tafj.home                          = C:\Temenos\TAFJ
temn.tafj.policy.file=<tafj.home>/conf/TAFJ.policy
```

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

```
# Specify the Precompile classes directories
# You can specify multiple directory, separated by
# ':' or ';' (eg /home/t24/lib;/home/t24/FT)
#
temn.tafj.directory.precompile=C:\Temenos\T24\bnk\t24lib
```

5. Set up a database connection.

```
#
# Database setup
#
#*********************************************************************

# URL that describes database connection,
# ex.  oracle:          jdbc:oracle:thin:@<host_or_ip>:1521:<db_name>
# ex.  db2:             jdbc:db2://<host_or_ip>:50000/<db_name>
# ex.  ms-sql:          jdbc:sqlserver://<host_or_ip>:1433;databaseName=<db_name>;integratedSecurity=true
# ex.  H2:              jdbc:h2:tcp://<host_or_ip>/<db_name>
# ex.  H2(Embedded):    jdbc:h2:<path_to_my_db>
# ex.  nuodb:           jdbc:com.nuodb://<host_or_ip>/<db_name>
#
temn.tafj.jdbc.url=jdbc:sqlserver://10.23.50.xxx:1433;databaseName=201905;integratedSecurity=false

# Class that describes the specific Driver for a database,
# ex. oracle:             oracle.jdbc.driver.OracleDriver
# ex. db2:                com.ibm.db2.jcc.DB2Driver
# ex. ms-sql:             com.microsoft.sqlserver.jdbc.SQLServerDriver
# ex. H2:                 org.h2.Driver
# ex. nuodb:              com.nuodb.jdbc.Driver
#
temn.tafj.jdbc.driver=com.microsoft.sqlserver.jdbc.SQLServerDriver

temn.tafj.jdbc.username=xxxx
temn.tafj.jdbc.password=xxxx
```

a. Under the Database setup header section, enter a value for `temn.tafj.jdbc.url`.

b. In `temn.tafj.jdbc.url` change `integratedSecurity` to false.

c. Provide the relevant driver information in `temn.tafj.jdbc.driver`.

d. Enter the database username and password.

6. Ensure that `temn.tafj.runtime.directory.current` points to the UD folder from `%T24_HOME%`.

```
# Specify what will be considered as the "current" directory (eg in an OPEN "." ...)
#
temn.tafj.runtime.directory.current=C:\Temenos\T24\bnk\UD
```

7. 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).

```
#
temn.tafj.jdbc.write.use.merge=false

# Use the MERGE statement instead of UPDATE/INSERT for NO XML Schema and NO XML Schema work
#
temn.tafj.jdbc.write.use.merge.no.xml=false
```

8. Set the value of `temn.tafj.locking.mode` to `DATABASE`.

```
Coherence supported (!! Experimental !!).
#
temn.tafj.locking.mode=DATABASE
```

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:** Refer to TAFJ-Lock `Manager.pdf` in *`$TAFJ_HOME`*`/doc` for information on creating and populating the TAFJ_HASHLOCKS table (which is required for DATABASE locking mode).

6.1.1 Verifying the TAFJ installation

Procedure

1. In *`%TAFJ_HOME%`*`/bin`, open the command prompt and enter `tDiag`. 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.

```
Home                    : 'C:\Temenos\TAFJ'
Conf directory          : 'C:\Temenos\TAFJ/conf'
Log directory           : 'C:\Temenos\TAFJ\log'
Log directory T24       : 'C:\Temenos\TAFJ\log_T24'
Log directory COMO      : 'C:\Temenos\TAFJ\log_T24\como'
Logging implementation  : LOG4J2
Version                 : DEV_201905
------------------------------------------------------------------

java.home               : C:\Temenos\3rdParty\java\jre
java.vendor             : Oracle Corporation
java.version            : 1.8.0_211
os.arch                 : amd64
os.name                 : Windows Server 2016
HostName                : R20-Stack-6
IP Address              : 10.23.50.212
Runtime                 : Double Byte SysSeparator
------------------------------------------------------------------

Default Project         : 'tafj'
------------------------------------------------------------------
Common validation:
Common validation completed - All fine
------------------------------------------------------------------

- Project : 'tafj'
       Java src dir           : C:\Temenos\TAFJ\data\tafj\java
       Java classes dir       : C:\Temenos\TAFJ\data\tafj\classes
       Precompile             : C:\Temenos\T24\bnk\t24lib
       Update dir             : C:\Temenos\TAFJ/updates
       ---------------------------------------------------------------
       Java default package    : com.temenos.t24
       ---------------------------------------------------------------
       DataBase URL           : jdbc:sqlserver://10.23.50.   :1433;databaseName=201905;integratedSecurity=false
       DataBase user          : tafj
       ---------------------------------------------------------------
       Locking mode           : DATABASE
       Locking name           :
       ---------------------------------------------------------------
       Current dir            : C:\Temenos\T24\bnk\UD
       Hold dir               : C:\Temenos\T24\bnk\UD/&HOLD&
       UD encoding            : UTF-8
       ---------------------------------------------------------------
       Timezone               :
       Local                  : en_US
       Debug enabled          : true
       ---------------------------------------------------------------
       JMS logger enabled     : false
       Logger API enabled     : true
       ---------------------------------------------------------------
       Session monitor enabled : true
       Session monitor host    : localhost
       Session monitor port    : 8377
       ------------------------------------------------------------
       Project validation:
       TAFJ Java Functions Version in DB   : 18.2.6
       TAFJ Java Functions Version in File : 18.2.6
       Java stored functions validated
       No duplicated classes found.
       Project validation completed - All fine
```

7 Sanity Check

Before the application server can be deployed, you must perform two sanity check to verify that the Transact standalone installation was successful.

7.1 Accessing Transact Classic

Transact classic is one of our user interfaces for Transact.

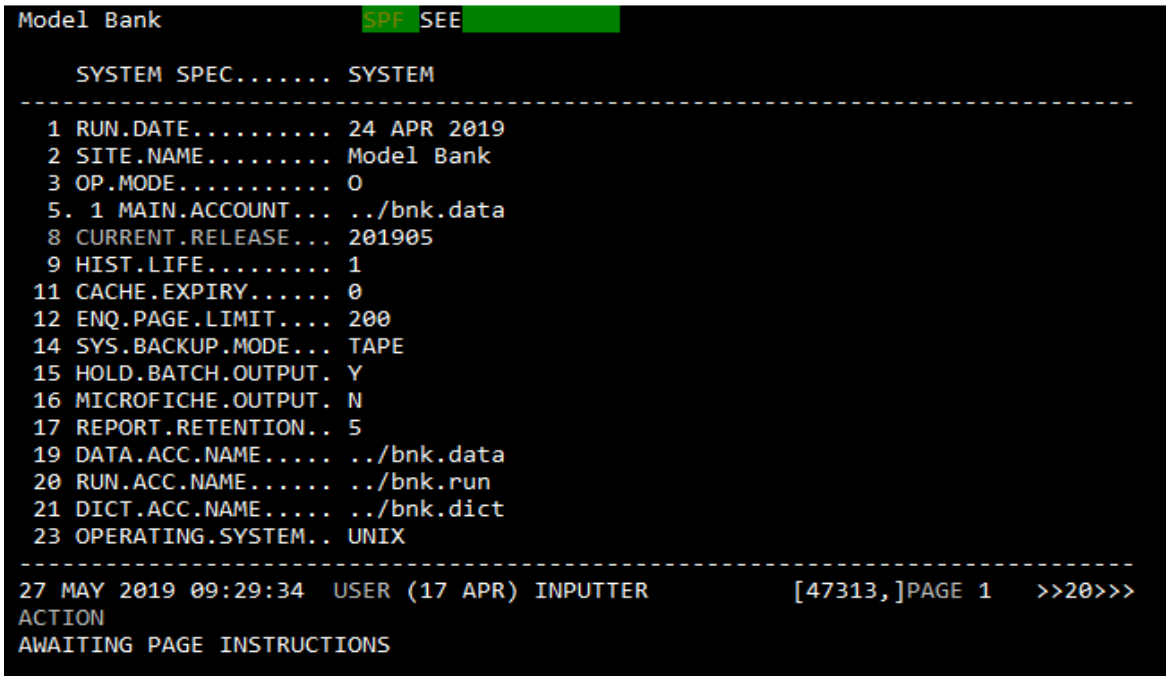
Procedure

1. In the File Explorer, open the command prompt from *%TAFJ_HOME%*\bin.

2. Enter tRun EX.

3. Login using a valid Transact user name and password.

4. Launch any Transact application. For example SPF.



7.2 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.

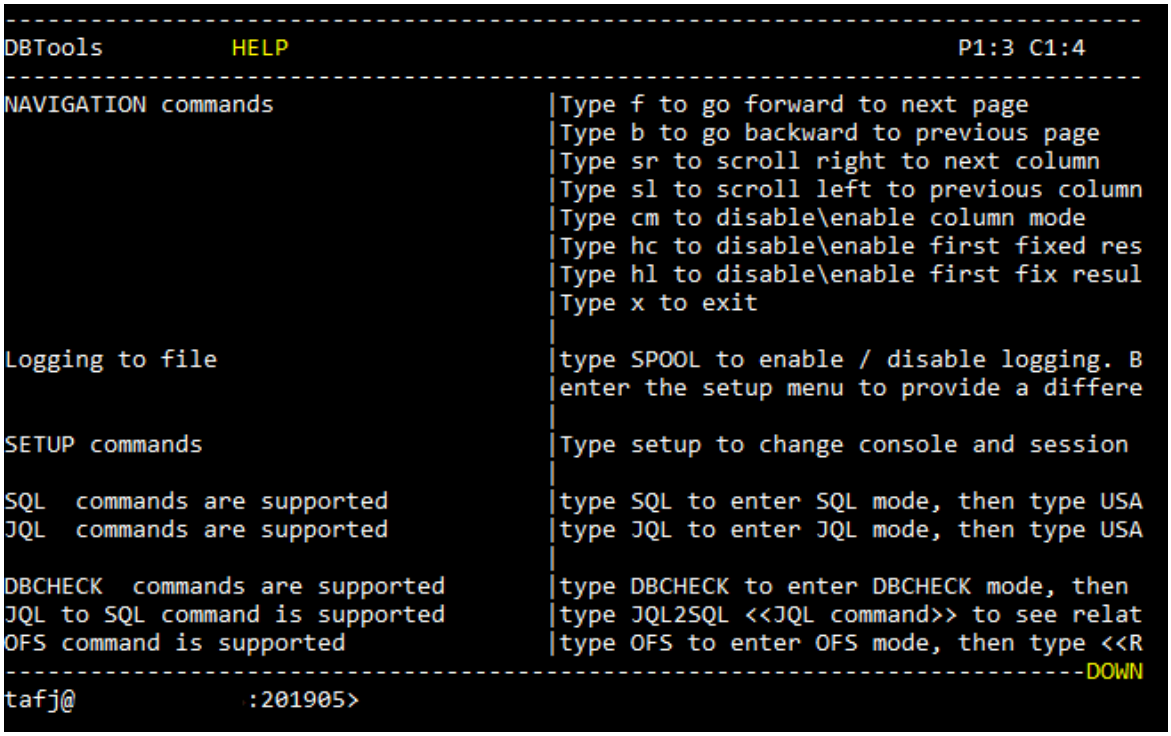
2. Create a TAFJ user by passing username and password parameters to the tUserMgnt.bat utility.

```
C:\Temenos\TAFJ\bin>tUserMgnt --Add -u tafjuser -p Temenos@123
```

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

```
C:\Temenos\TAFJ\bin>DBTools -u tafjuser -p Temenos@123
```

4. The DBTools console is launched. View the database name, database username and IP address displayed in the bottom left of the screen.



8 Configuring JBoss EAP 7.2

This section details how to configure JBoss modules for TAFJ/Transact and the database.

8.1 Configuration prerequisites

Before you start configuring JBoss with TAFJ/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.

8.1.1 Memory settings

The default memory setting is 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.

```
rem # JVM memory allocation pool parameters - modify as appropriate.
set "JAVA_OPTS=-Xms4G -Xmx6G -XX:MetaspaceSize=96M -XX:MaxMetaspaceSize=1024m"
```

> **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.

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

```
rem # TAFJ Home and file encoding
set "JAVA_OPTS=%JAVA_OPTS% -Dtafj.home=%TAFJ_HOME% -Dfile.encoding=UTF-8"
```

8.2 Configuring JBoss modules

JBoss EAP 7.2 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 7.2 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 module. You need to create three modules on your application server.

- MSSQL Database driver module.
- TAFJ module.
- Transact module.

8.2.1 Deploying the MSSQL database driver module

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

The following procedure shows you how to deploy the MSSQL database driver/ module.

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.

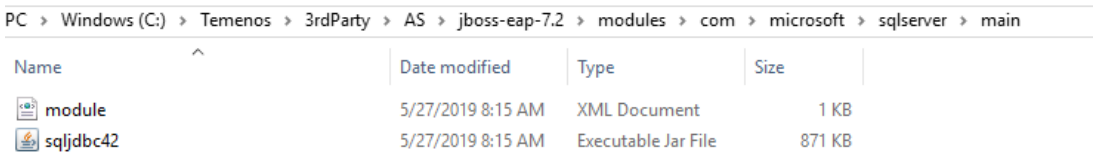
```
mkdir com\microsoft\sqlserver\main
```

3. Copy `module.xml` from *`%TAFJ_HOME%`* to the main folder for the corresponding database.

```
copy %TAFJ_
HOME%\appserver\jboss\jboss7eap\modules\com\microsoft\sqlserver\ma
in\module.xml %JBOSS_HOME%\modules\com\temenos\tafj\main
```

4. Copy the corresponding database driver to the main folder of sqlserver.

```
copy %TAFJ_HOME%\ext\sqljdbc42.jar %JBOSS_
HOME%\modules\com\microsoft\sqlserver\main
```



5. Open `module.xml` to check the jar name of driver matches the folder path for that module.

```
<?xml version="1.0" encoding="UTF-8"?>

<module xmlns="urn:jboss:module:1.1" name="com.microsoft.sqlserver">

    <resources>
        <resource-root path="sqljdbc42.jar"/>
    </resources>
    <dependencies>
        <module name="javax.api"/>
        <module name="javax.transaction.api"/>
        <module name="javax.servlet.api" optional="true"/>
        <module name="javax.xml.bind.api"/>
```

8.2.2 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.

```
mkdir temenos\tafj\main
```

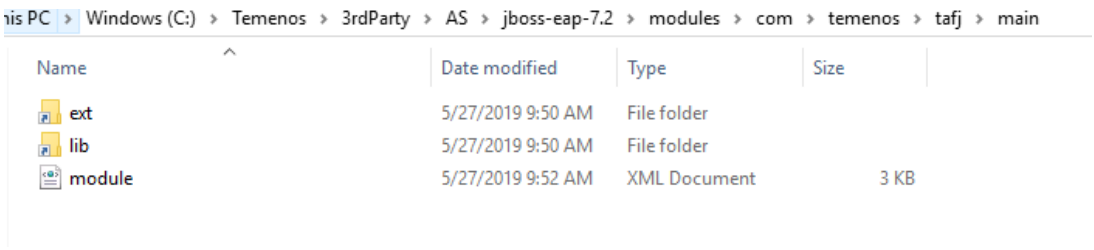
2. Copy the tafj module from *%TAFJ_HOME%* to the main folder to the *%JBOSS_HOME%*\modules folder.

```
copy %TAFJ_
HOME%\appserver\jboss\jboss7eap\modules\com\temenos\tafj\main\modu
le.xml %JBOSS_HOME%\modules\com\temenos\tafj\main
```

3. In *%JBOSS_HOME%*\modules\com\temenos\tafj\main open the command prompt.

4. Create a symbolic link to *%TAFJ_HOME%*\lib and *%TAFJ_HOME%*\ext.

```
mklink /D lib %TAFJ_HOME%\lib
mklink /D ext %TAFJ_HOME%\ext
```



5. Open module.xml. The TAFJ module has dependencies on the Transact module that you

need to create. It also contains the MSSQL database driver module. Uncomment the MSSQL database driver module dependency in the <dependencies> section.

```
<dependencies>
  <module name="com.temenos.t24"/>
  <!--
  <module name="com.oracle.ora12c"/>
  <module name="com.ibm.db2v11"/>
  <module name="com.microsoft.sqlserver"/>
  <module name="com.nuodb.jdbc"/>
   -->
  <module name="com.microsoft.sqlserver"/>
  <module name="javax.api"/>
  <module name="javaee.api"/>
```

8.2.3 Configuring the Transact module

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

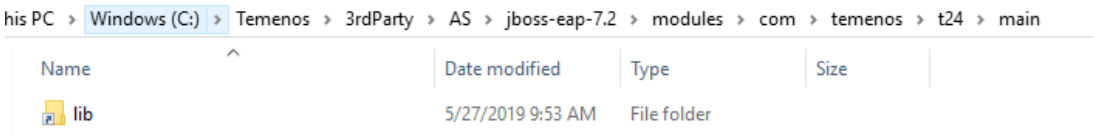
Procedure

1. In the File Explorer, in the temenos folder create this directory structure.

```
mkdir t24\main
```

2. Open command prompt from *%JBOSS_HOME%*\modules\com\temenos\t24\main.

3. Create a symbolic link to *%T24_HOME%*\t24lib.



8.2.4 Generating the Transact module.xml

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

Procedure

1. Create the file in *%TAFJ_HOME%*\bin.

```
JBossTools module_name path_to_jars dest [root_prefix] [-tafjdep]
```

Options	Description
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 %T24_HOME%\t24lib %JBOSS_
HOME%\modules\com\temenos\t24\main lib -tafjdep
```

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

```
  </resources>
  <dependencies>
    <module name="com.temenos.tafj"/>
  </dependencies>
</module>
```

9 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 standalone installation.

Procedure

1. In the File Explorer, copy and paste standalone-full.xml into *%JBOSS_HOME%*\standalone\configuration and rename it to T24.xml.

2. Start JBoss using the command given below.

```
%JBOSS_HOME%\bin\standalone.bat -b 0.0.0.0 -bmanagement 0.0.0.0 --
server-config=T24.xml
```

> **NOTE:** Note that as we are using T24.xml, this command must use the option --server-config to specify the configuration file.

   To deploy Transact on TAFJ on EAP, the various subsystems in T24.xml must be configured. TAFJ uses the management Command Line Interface (CLI) interface tool to automate the process of configuring the J2EE profiles.

   JBoss CLI can be launched from *%JBOSS_HOME%*\bin folder by running jboss-cli.bat script. It takes two arguments, --file and --properties.

   The --file argument enables CLI commands to be provided from a text file. *%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.

3. Navigate to *%TAFJ_HOME%*\appserver\jboss\jboss7eap\jboss-cli. Open T24Setup.cli.

4. In the drivers section, uncomment the driver parameters according to the database used.

```
#Drivers
#define the driver corresponding to your database - H2 drive
corresponding to your database
#please note that you have to create the module for the driv
#/subsystem=datasources/jdbc-driver=ora12c:add(driver-name=c
client.OracleXADataSource)
/subsystem=datasources/jdbc-driver=sqljdbc:add(driver-name=s
osoft.sqlserver.jdbc.SQLServerXADataSource)
```

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

```
#System property
#/system-property=tafj.home:add(value=${TAFJ_HOME})
/system-property=file.encoding:add(value=UTF-8)
```

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`.

6. Open `tafj.properties`.

7. Enter values for the `TAFJ_HOME`, `DB_URL`, `DB_DRIVER`, `DB_USER`, `DB_PWD` parameters.

```
TAFJ_HOME=C:\Temenos\TAFJ
#############################################################
# database details
#
#DB_URL doesn't apply for DB2 XA DRIVER
DB_URL="jdbc:sqlserver://10.23.50.xxx:1433;databaseName=201905;integratedSecurity=false"
DB_DRIVER=sqljdbc
#DB_DRIVER=[h2,ora12c,db2jdbc,sqljdbc,nuodb-jdbc]
DB_DRIVER_XA=[h2,ora12cXA,db2jdbc,sqljdbc,nuodb-jdbc]
DB_USER=xxx
DB_PWD=xxx
#DB2 XA DRIVER in replacement of DB_URL
DB_HOST=[hostname]
DB_NAME=[instance name]
DB_PORT=[port]
```

8. 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 `<validate-operation-requests>` is set to `true`.

```
<validate-operation-requests>true</validate-operation-requests>

<!-- Command and operation history log configuration -->
<history>
    <enabled>true</enabled>
    <file-name>.jboss-cli-history</file-name>
    <file-dir>${user.home}</file-dir>
    <max-size>500</max-size>
</history>

<!-- whether to resolve system properties specified as command a
                in the CLI VM before sending the operation request
<resolve-parameter-values>true</resolve-parameter-values>
```

9. Open a cmd prompt from location *%JBOSS_HOME%*/bin. Run the jboss-cli command.

```
%JBOSS_HOME%\bin\jboss-cli.bat --connect
--file=C:\Temenos\TAFJ\appserver\jboss\jboss7eap\jboss-
cli\T24Setup.cli
--properties=C:\Temenos\TAFJ\appserver\jboss\jboss7eap\jboss-
cli\tafj.properties
```

On successful execution the command updates the required subsystems in T24.xml (which was used to startup JBoss). If this is successful, you will see text similar to the illustration below.

```
C:\Temenos\3rdParty\AS\jboss-eap-7.2\bin>%JBOSS_HOME%\bin\jboss-cli.bat --connect --file=C:\Temenos\TAFJ\appserver\jboss
\jboss7eap\jboss-cli\T24Setup.cli --properties=C:\Temenos\TAFJ\appserver\jboss\jboss7eap\jboss-cli\tafj.properties
The batch executed successfully
process-state: reload-required
Press any key to continue . . .
```

The scripts now add the following items to the standalone configuration file (T24.xml):

- System properties
- Drivers
- Data sources
- Global modules (TAFJ and Transact)

- JMS Pools
- JMS Queues

10. Stop the JBoss instance (as the configuration of JBoss with Transact-TAFJ is now complete).

10 Extracting UXP browser artefacts

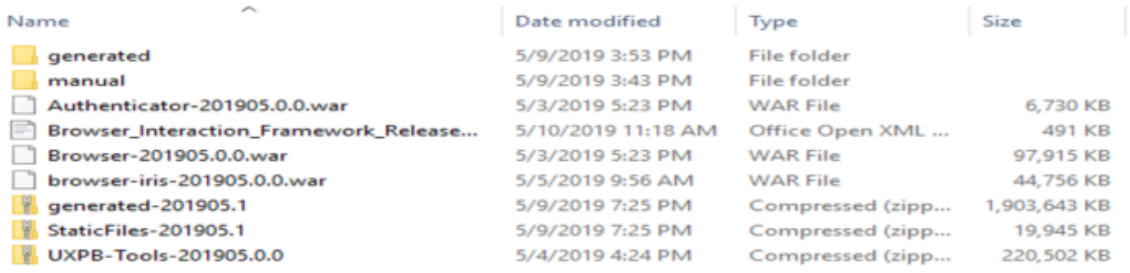
This section deals with the steps necessary before deploying Transact and TAFJ artefacts. It is only required if UXP Browser is to be deployed.

Procedure

1. Obtain `UXP-Browser.201905.zip` from Distribution and copy it to the `%TEMENOS_HOME%\UXPBrowser` folder.

2. In this folder unzip `UXP-Browser.201905.zip`.

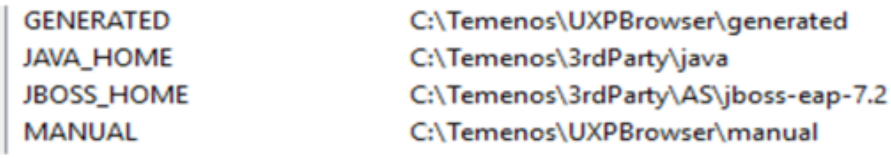
3. The extracted file (`UXP-Browser.201905.zip`) file contains `generated-201905.1.zip`. Extract `generated-201905.1.zip` to obtain the `manual` and `generated` sub-folders.



Name	Date modified	Type	Size
generated	5/9/2019 3:53 PM	File folder	
manual	5/9/2019 3:43 PM	File folder	
Authenticator-201905.0.0.war	5/3/2019 5:23 PM	WAR File	6,730 KB
Browser_Interaction_Framework_Release...	5/10/2019 11:18 AM	Office Open XML ...	491 KB
Browser-201905.0.0.war	5/3/2019 5:23 PM	WAR File	97,915 KB
browser-iris-201905.0.0.war	5/5/2019 9:56 AM	WAR File	44,756 KB
generated-201905.1	5/9/2019 7:25 PM	Compressed (zipp...	1,903,643 KB
StaticFiles-201905.1	5/9/2019 7:25 PM	Compressed (zipp...	19,945 KB
UXPB-Tools-201905.0.0	5/4/2019 4:24 PM	Compressed (zipp...	220,502 KB

4. Set up the environment variables for the `generated` and `manual` sub-folders.

   - `GENERATED=%UXPBROWSER%\generated`
   - `MANUAL=%UXPBROWSER%\manual`



GENERATED	C:\Temenos\UXPBrowser\generated
JAVA_HOME	C:\Temenos\3rdParty\java
JBOSS_HOME	C:\Temenos\3rdParty\AS\jboss-eap-7.2
MANUAL	C:\Temenos\UXPBrowser\manual

5. The war files can be deployed without renaming. If necessary you can rename them from

`Browser-201905.0.0.war`,`browser-iris-201905.0.0.war` and `Authenticator-201905.0.0.war` to `Browser.war`,`browser-iris.war` and `Authenticator.war`.

11 Deploying Transact and TAFJ artefacts

The final step in the setup is to deploy the Transact Product UI components UXP Browser (UXPB) and BrowserWeb. UXPB is our latest user interface for working with Transact. BrowserWeb is its antecedent. This runbook does not cover deployment of our other product user interfaces like Temenos Connect Internet Banking (TCIB) and Mobile Banking (TCMB).

11.1 Deploying TAFJEE and TAFJ spooler plugins

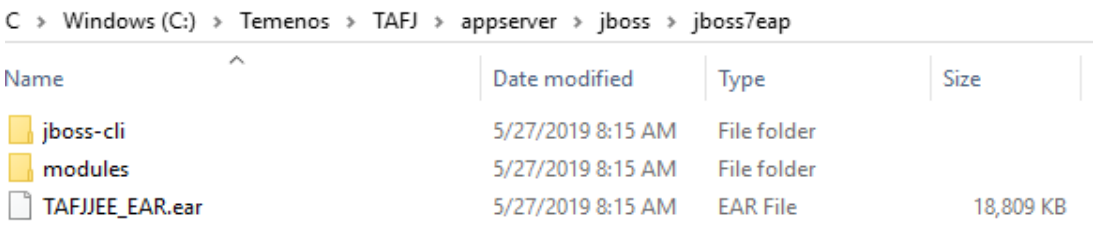
TAFJ-JEE application is the connecting link between client applications such as BrowserWeb and Transact. The TAFJ-JEE application is named TAFJJEE_EAR.ear. This ear file comprises of channel-specific MDBs (`TAFJJEE_MDB.jar`) and EJBs (`TAFJJEE_EJB.jar`) to read messages from JMS Queues, call Transact and publish responses in reply queues.

TAFJ uses the TAFJ spooler plugin to redirect reports destined for the UD directory to another machine or report repository. `TAFJJEE_EAR.ear` fails to deploy with an error if `PrintHold` is missing, or if `TAFJSpoolerPlugins.rar` is not present.

Procedure

1. In the File Explorer, copy `TAFJSpoolerPlugins.rar` from `%TAFJ_HOME%\appserver` to the deployments folder.

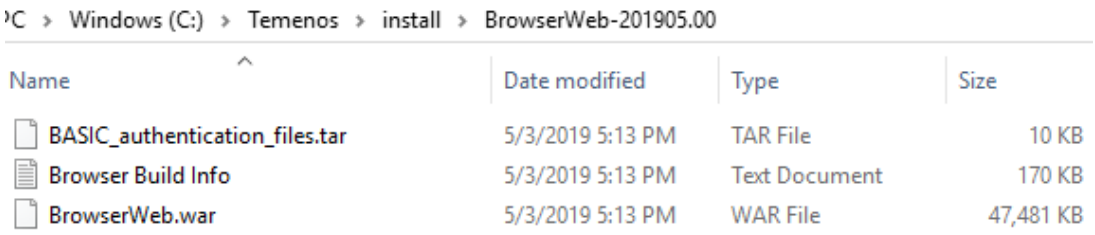
2. Copy `TAFJJEE_EAR.ear` from `%TAFJ_HOME%\appserver\jboss\jboss7eap` to the deployments folder.



11.2 Deploying BrowserWeb

Procedure

1. In file explorer, copy the `BrowserWeb-201905.00.tar` file from distribution to `%TEMENOS_HOME%\install` folder.
2. Extract the `BrowserWeb-201905.00.tar` file in the install folder.
3. Copy the `BrowserWeb.war` file and paste it in the `%JBOSS_HOME%\standalone\deployments` folder.



11.3 Generating the keystore files

JWT token-based authentication is used with UXPB. When a request comes from UXPB, the request must first be authenticated. User credentials are sent to the authentification service (i.e.`Authenticator.war`) to verify the credentials. When the credentials are successfully verified, Authentification sends back a signed JWT token containing the user information. The IRIS service (i.e. `browser-iris.war`) that processes the request validates the JWT token and extracts the user information before handling the request.

In order to do all this, we need a pair of keys (either RSA private keys or RSA public keys). Follow the below steps to generate a keystore file and then export the public key into a certificate file.

> **NOTE:** In this document we have used Java's default keystore type JKS.

Procedure

1. In the File Explorer go to *$JAVA_HOME*/bin.

2. Create a .jks keystore file.

   ```
   keytool -genkey -alias temenos_uxpb -keyalg RSA -
   keystore uxpb.jks -keysize 2048
   ```

   Note that:

   - uxpb.jks is the name of the new file. It contains one RSA key.
   - temenos_uxpb is the name of the alias.
   - RSA is the algorithm that creates the key.
   - 2048 is the size of the key in bits.
   - JKS is the default format of the KeyStore.

   Press Enter after entering the text above.

3. Enter the following information when prompted:

   a. The password for the keystore.

   Look in Browser.war/WEB-INF/classes/authServiceContext.xml to find the password. Alternatively, locate it in Authenticator.war or browser-iris.war.

   To change the password, change it in authServiceContext.xml in Browser.war, Authenticator.war and browser-iris.war.

```
<bean id="JwTokenConfig" class="com.temenos.technology.t24Security.JwTokenConfig">
    <property name="issuer" value="T24" />
    <property name="tokenTimeoutMinutes" value="3600" />
    <property name="principalClaim" value="sub" />
    <property name="jksFilePath" value="uxpb.jks" />
    <property name="jksKeystorePass" value="Temen0s" />
    <property name="jksAlias" value="temenos_uxpb" />
```

b. Re-type the password.

c. Your first and last name.

d. The organisation unit name.

e. Re-type the organisation unit name.

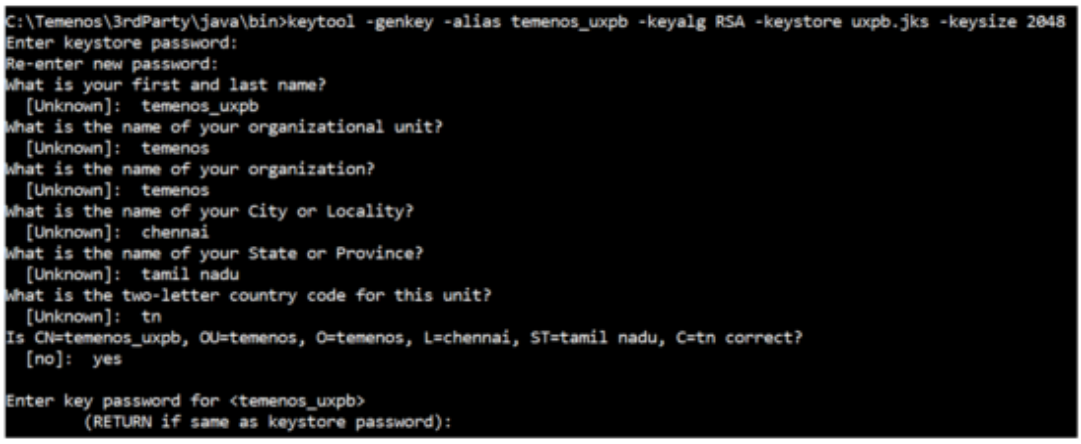
f. The city name.

g. The state or province.

h. Two characters for the country.

i. Enter Yes to the last prompt.

j. When prompted `Enter key password for temenos_uxpb` just press Enter.



4. Now that you have a KeyStore file, the next step is to export the public key as a separate entity. Create a `.cer` keystore file.

```
keytool -export -keystore uxpb.jks -alias temenos_uxpb
-file temenos_uxpb.cer
```

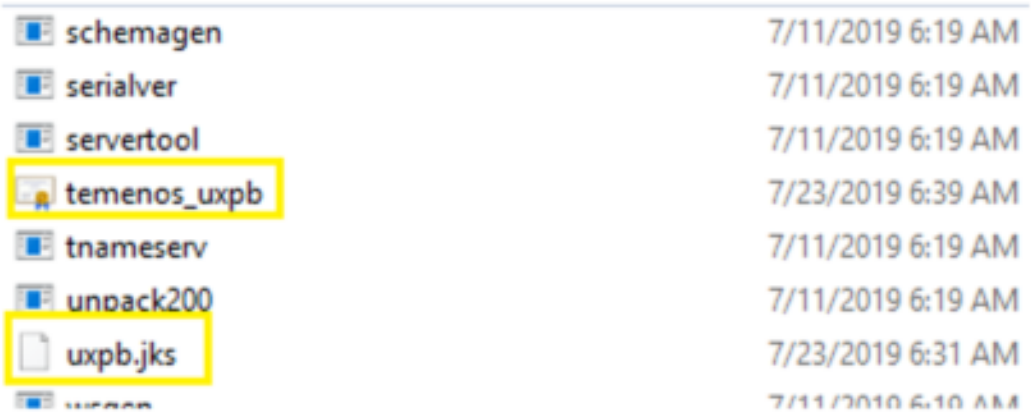
This command specifies the jks file, the alias of the key, and the output file `temenos_uxpb.cer`.

5. Enter the password. This can be found in `authServiceContent.xml`.

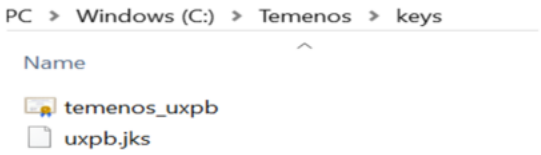
   The `cer` file is now created.

```
C:\Temenos\3rdParty\java\bin>keytool -export -keystore uxpb.jks -alias temenos_uxpb -file temenos_uxpb.cer
Enter keystore password:
Certificate stored in file <temenos_uxpb.cer>
```

   In *$JAVA_HOME*`/bin` the `.jks` and `.cer` files can now be seen.



6. Create a directory anywhere as you wish (for example a directory called `keys` in the `Temenos` directory) and move the `.jks` and `.cer` files to this directory.



11.4 Deploying UXP Browser

The UXP Browser artefact is the Browser. UXPB requests are routed through Interaction framework. Therefore, along with the Browser, you must also deploy the browser-iris application that contains the interaction resources.

> **NOTE:** This section is only required if you have processed "Extracting UXP browser artefacts" on page 40.

Procedure

1. In the File Explorer, open `browser-iris.war` as an achive and go to `/WEB-INF/classes`.
2. Delete `temenos_uxpb.cer`.
3. Open `authServiceContext.xml` and add the path to the `.cer` file.

4. Copy `browser-iris.war` from the UXPBrowser folder to *`%JBOSS_HOME%`*`\standalone\deployments`.
5. Open `Browser.war` as an archive and go to `/WEB-INF/classes`.

6. Delete `temenos_uxpb.cer` and `uxpb.jks`.

7. Open `authServiceContext.xml` and add the paths to the `.jks` and `.cer` files.

```
    <bean id="JwTokenConfig" class="com.temenos.technology.t24Security.JwTokenConfig">
        <property name="issuer" value="T24" />
        <property name="tokenTimeoutMinutes" value="3600" />
        <property name="principalClaim" value="sub" />
        <property name="jksFilePath" value="C:\Temenos\keys\uxpb.jks" />
        <property name="jksKeystorePass" value="Temen0s" />
        <property name="jksAlias" value="temenos_uxpb" />
        <property name="publicKeyCertPath" value="C:\Temenos\keys\temenos_uxpb.cer" />
    </bean>
</beans>
```

8. Copy `browser-iris.war` from the UXPBrowser folder to *`%JBOSS_HOME%`*`\standalone\deployments-folder`.

11.5 Deploying Authenticator

This war file is required for authentication of UXP Browser. Without this, you will not be able to login into UXP Browser. This section is not applicable if you have not processed "Deploying UXP Browser" on the previous page.

Procedure

1. In File Explorer open `Authenticator.war` as an archive and go to `/WEB-INF/classes`.

2. Delete `temenos_uxpb.cer` and `uxpb.jks`.

3. Open `authServiceContext.xml` and add the path to the `.jks` file.

```
    <bean id="JwTokenConfig" class="com.temenos.technology.t24Security.JwTokenConfig">
        <property name="issuer" value="T24" />
        <property name="tokenTimeoutMinutes" value="3600" />
        <property name="principalClaim" value="sub" />
        <property name="jksFilePath" value="C:\Temenos\keys\uxpb.jks" />
        <property name="jksKeystorePass" value="Temen0s" />
        <property name="jksAlias" value="temenos_uxpb" />
        <property name="publicKeyCertPath" value="temenos_uxpb.cer" />
    </bean>
</beans>
```

4. Copy `Authenticator.war` from the UXPBrowser folder to *%JBOSS_HOME%*`\standalone\deployments`.

12 Deploying Transact component services

Procedure

1. In the Transact distribution pack, in the `extensions` folder, locate the Transact component service APIs. These will be Axis2 archives named `t24-`*`ComponentService`*`-jws.aar`
2. Deploy these archives into the Apache Axis2 web archive file.

> **NOTE:**
> The release version of Apache Axis2 used here is 1.6.2.

This section tells you how to deploy `axis2.war` on the application server.

12.1 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` as an archive and navigate to `WEB-INF/lib`.
5. Add the following dependency JARs to the `lib` folder.
   - `commons-pool-1.5.5.jar`
   - `spring-2.5.6.jar`
6. Copy the component service archives from *`%T24_HOME%`*`/Extensions` into the `/WEB-INF/services` directory of `axis2.war`.

7. Open `/WEB-INF/services/services.list` and add the names of the component service archives that you added to the services directory.

```
t24-EB_CatalogService-tafj-jws.aar
t24-IF_IntegrationLandscapeService-tafj-jws.aar
t24-DS_DesignStudioInstallerService-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
version-1.6.2.aar
```

8. Open `/WEB-INF/web.xml` and configure database connectivity by adding the data source.

```
<!-- Temenos T24 Service Provider Specific Settings -->
       <resource-ref id="ResourceRef_t24DataSource">
              <description>Used to get connections from T24
              jdbc pool
              </description>
              <res-ref-name>jdbc/t24DataSource</res-ref-name>
              <res-type>javax.sql.DataSource</res-type>
              <res-auth>Container</res-auth>
       </resource-ref>
       <resource-ref id="ResourceRef_t24LockingDataSource">
              <description>Used to get connections from T24
               jdbc pool
              </description>
              <res-ref-name>jdbc/t24LockingDataSource
              </res-ref-name>
              <res-type>javax.sql.DataSource</res-type>
              <res-auth>Container</res-auth>
</resource-ref>
```

```
        <error-code>500</error-code>
        <location>/axis2-web/Error/error500.jsp</location>
    </error-page>

    <!-- Temenos T24 Service Provider Specific Settings -->
    <resource-ref id="ResourceRef_t24DataSource">
        <description>Used to get connections from T24 jdbc pool
        </description>
        <res-ref-name>jdbc/t24DataSource</res-ref-name>
        <res-type>javax.sql.DataSource</res-type>
        <res-auth>Container</res-auth>
    </resource-ref>
    <resource-ref id="ResourceRef_t24LockingDataSource">
        <description>Used to get connections from T24 jdbc pool
        </description>
        <res-ref-name>jdbc/t24LockingDataSource</res-ref-name>
        <res-type>javax.sql.DataSource</res-type>
        <res-auth>Container</res-auth>
    </resource-ref>
</web-app>
```

9. In `WEB-INF` add the the jboss specific deployment descriptor (`jboss-web.xml`).

10. Configure database connectivity by adding the data sources

```
<?xml version="1.0" encoding="UTF-8"?>
<jboss-web>
        <resource-ref>
                <res-ref-name>jdbc/t24DataSource</res-ref-name>
                <res-type>javax.sql.DataSource</res-type>
                <jndi-name>java:/jdbc/t24Axis2DS</jndi-name>
        </resource-ref>
        <resource-ref>
                <res-ref-name>jdbc/t24LockingDataSource
                </res-ref-name>
                <res-type>javax.sql.DataSource</res-type>
                <jndi-name>java:/jdbc/t24LockingDS</jndi-name>
        </resource-ref>
<jboss-web>
```

```
<?xml version="1.0" encoding="UTF-8"?>
<jboss-web>
    <resource-ref>
        <res-ref-name>jdbc/t24DataSource</res-ref-name>
        <res-type>javax.sql.DataSource</res-type>
        <jndi-name>java:/jdbc/t24Axis2DS</jndi-name>
    </resource-ref>
    <resource-ref>
        <res-ref-name>jdbc/t24LockingDataSource</res-ref-name>
        <res-type>javax.sql.DataSource</res-type>
        <jndi-name>java:/jdbc/t24LockingDS</jndi-name>
    </resource-ref>
</jboss-web>
```

11. Save the changes to `axis2.war`.

12.2 Deploying axis2

Procedure

1. In the File Explorer, navigate to *%JBOSS_HOME%*`/standalone/configuration`.
2. Open `T24.xml`.
3. In `datasource`, add the data source for `axis2.war` deployment.

```
<datasource jta="true" judo-name="java:/jdbc/t24Axis2DS" pool-
name="Axis2DS" enabled="true" use-java-context="true" use-
ccm="true">
<connection-
url>jdbc:sqlserver://10.23.50.xxx:1433;databaseName=201905;integra
tedSecurity=false</connection-url>
<connection-property name="defaultRowPrefetch">
                100
        </connection-property>
        <driver>sqljdbc</driver>
        <pool>
```

```
                <min-pool-size>5</min-pool-size>
                <max-pool-size>220</max-pool-size>
                <flush-strategy>FailingConnectionOnly
                </flush-strategy>
        </pool>
        <security>
                <user-name>xxx</user-name>
                <password>xxx</password>
        </security>
</datasource>
```

4. Copy `axis2.war` from UXPBrowser folder to *`%JBOSS_HOME%`*`/standalone/deployments`.

   `cp axis2.war` *`%JBOSS_HOME%`*`/standalone/deployments`

13 Sanity check - post deployment

Login to Transact through the product user interfaces to verify your application server deployment. This section describes the steps to login using UXPB and BrowserWeb.

13.1 Starting the application server

Procedure

1. In *%JBOSS_HOME%*\bin use `standalone.bat` to start JBoss.

   The parameters required depend on the artefacts being deployed. If you are not deploying UXP browser (as mentioned in "Extracting UXP browser artefacts" on page 40, "Deploying UXP Browser" on page 47 and "Deploying Authenticator" on page 48) the following are sufficient.

   ```
   %JBOSS_HOME%\bin\standalone.bat --server-config=T24.xml
   -b 0.0.0.0 -Djboss.http.port=9089
   ```

   If you are deploying UXP Browser, the following are required.

   ```
   %JBOSS_HOME%\bin\standalone.bat --server-config=T24.xml
    -Djboss.http.port=9089 -Djboss.node.name=node1 -b 0.0.0.0
    -Djboss.as.management.blocking.timeout=1200
    -Dcom.edgeipk.librarypath=%MANUAL%/edge,%GENERATED%/edge
    -
   DedgeSystemPropertyUpdaters=com.temenos.connect.system.BRPRuntimeP
   roperties
    -Doverride.IRIS_
   MAP=%MANUAL%/edge/data/HrefComponentManualMap.properties,%GENERATE
   D%/edge/data/HrefComponentMap.properties
    -Dcom.temenos.interaction.config.dir=%GENERATED%/iris-
   Doverride.SLANG_
   FOLDERS=%MANUAL%/edge/data/slang/manual,%GENERATED%/edge/data/slan
   g/generated,$$PROJECHOME$/WEB-
   INF/data/slang/default -Dcom.edgeipk.checkForUpdates=Y
    -Doverride.ENABLE_COMMAND_LINE=Y -
   ```

```
Diris.cache.index.file=%TEMENOS_HOME%\workspace\BRP-
models-gen\lastChange -Diris.skip.validation=true -
Dorg.apache.tomcat.util.buf.UDecoder.ALLOW_ENCODED_
SLASH=true -
Dorg.apache.catalina.connector.CoyoteAdapter.ALLOW_
BACKSLASH=true -DRTP_REAPER_MEM_USED_
PERCENT_ABORT=99 -Djava.awt.headless=true
```

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.

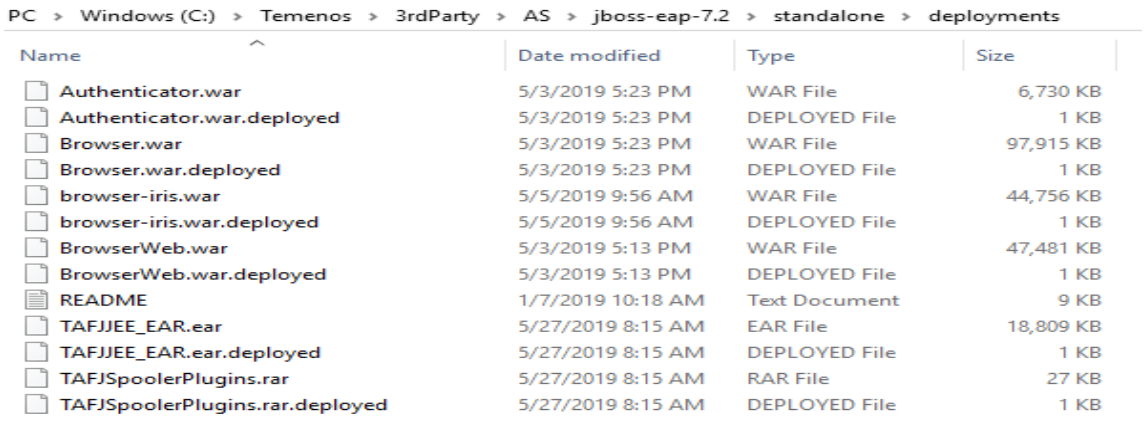
The arguments are explained in the table below.

Argument	Explanation
`--server-config`	The standalone profile configuration file to use for JBoss startup (e.g. `T24.xml`).
`-Djboss.http.port`	The port number of the application server.
`-Djboss.node.name`	A unique identifier for the application server instance.
`-D jboss.as.management.blocking.timeout`	A timeout value for container stability. If this value is reached during startup then all applications are un-deployed and the container is shutdown.
`-Dcom.edgeipk.librarypath`	The path to the UXP resources that need to be loaded (e.g. `/BRP/manual/edge`, `/BRP/generated/edge`).

Argument	Explanation
-DedgeSystemPropertyUpdaters	The properties to be loaded. One of the following options should be used: • com.temenos.connect.system.BRPRuntimeProperties • com.temenos.connect.system.BRPPreSalesProperties • com.temenos.connect.system.BRPSystemTestProperties • com.temenos.connect.system.BRPProductionProperties • com.temenos.connect.system.BRPDeveloperProperties
-Doverride.IRIS_MAP	The path to the two mapping files. • HrefComponentManualMap.properties provides mapping for manual adopted ifp's • HrefComponentMap.properties provides mapping for generated ifps
-Doverride.SLANG_FOLDERS	The path to the language files. The MANUAL file path precedes the GENERATED file path.
-Dcom.edgeipk.checkForUpdate	This determines whether an ifp is reread if it has been changed. If set to Y, it is reread. This is used in developer mode, when a change to an ifp does not require a server restart.
-Doverride.ENABLE_COMMAND_LINE	This determines whether to enable the command line for all users, irrespective of user profile configuration. If set to Y the command line will be enabled.

Argument	Explanation
-Diris.skip.validation	This determines whether hypermedia(rim) validation is enabled or disabled. It can be set to true or false.
-DRTP_REAPER_MEM_USED_PERCENT_ABORT	This specifies the percentage of memory used before preventing load of new contexts.

2. When the server is started, in the *%JBOSS_HOME%*\standalone\deployments folder check that all the artefacts listed have been deployed.



13.1.1 Additional modifications for release 201910 onwards

For version 201910 and onwards, some additional arguments are required to access UXP Browser. In addition to the arguments specified above, add the following arguments to the JBoss script.

```
-
DedgeSystemPropertyUpdaters=com.temenos.connect.system.BRPSystemTestPr
operties
-DBRP_HOME=C:\Temenos
```

13.2 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.

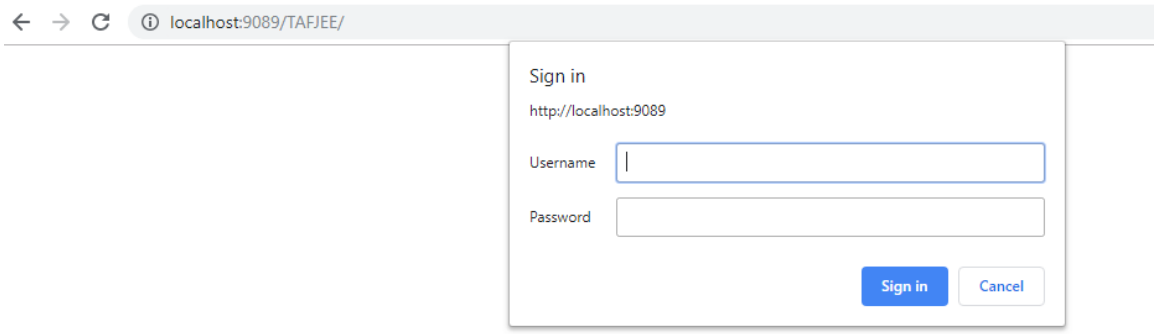
Procedure

1. Open the command prompt in *%JBOSS_HOME%*`/bin`.
2. Run add-user.bat to create an application user.
3. Add the user to the TAFJAdmin role. Respond as described below to the prompts.

Parameter	Value
Type of user.	Application user.
User name.	Your preferred user name.
Password.	Your preferred password.
Groups the user should belong to.	TAFJAdmin.
Add user for realm Application Realm.	Yes.
Final question (Is this new user going to....).	No.

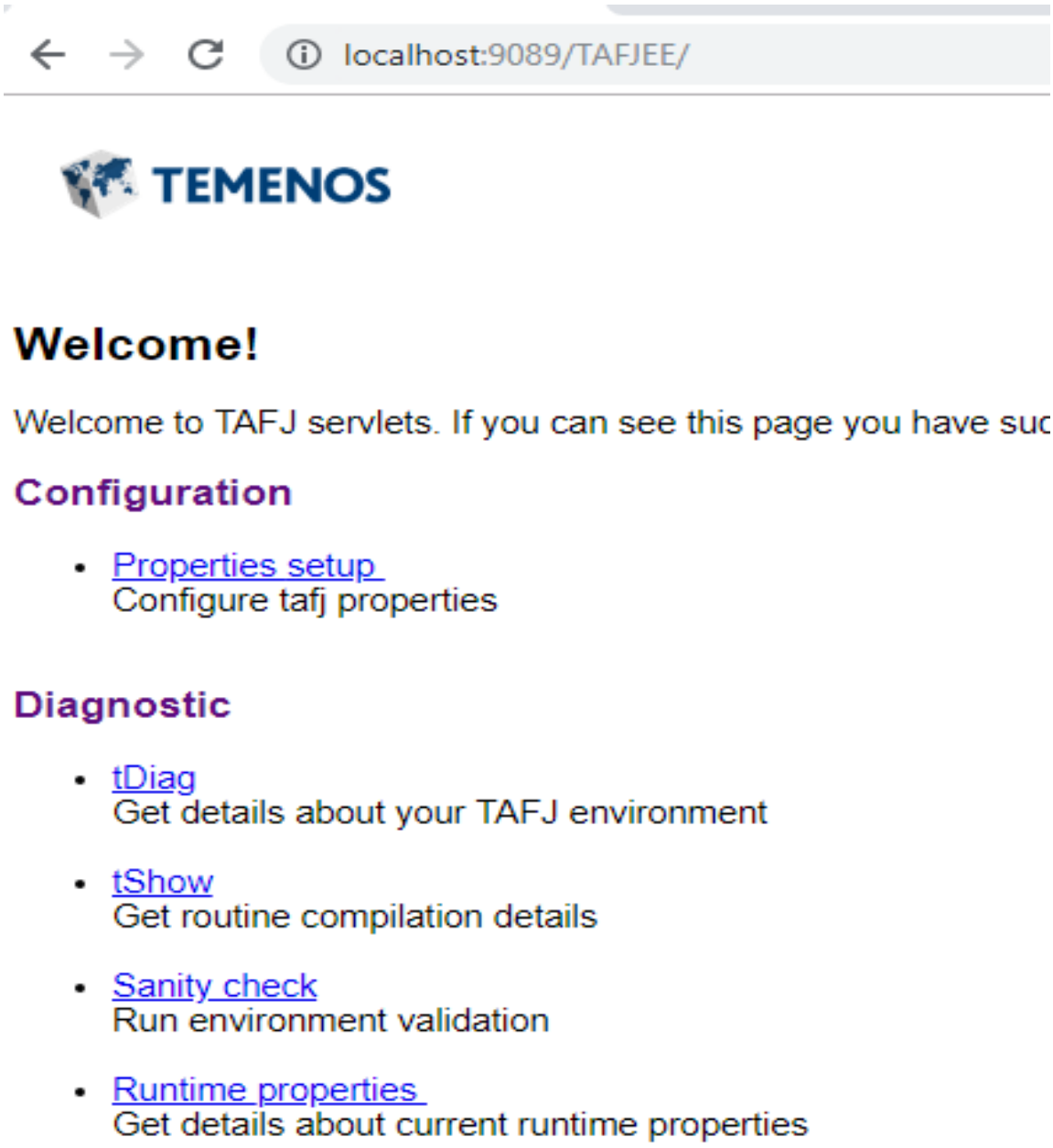
```
C:\Temenos\3rdParty\AS\jboss-eap-7.2\bin>add-user.bat

What type of user do you wish to add?
 a) Management User (mgmt-users.properties)
 b) Application User (application-users.properties)
(a): b

Enter the details of the new user to add.
Using realm 'ApplicationRealm' as discovered from the existing property files.
Username : TAFJUser
Password recommendations are listed below. To modify these restrictions edit the add-user.properties configuration file.
 - The password should be different from the username
 - The password should not be one of the following restricted values {root, admin, administrator}
 - The password should contain at least 8 characters, 1 alphabetic character(s), 1 digit(s), 1 non-alphanumeric symbol(s)
Password :
Re-enter Password :
What groups do you want this user to belong to? (Please enter a comma separated list, or leave blank for none)[  ]: TAFJAdmin
About to add user 'TAFJUser' for realm 'ApplicationRealm'
Is this correct yes/no? yes
Added user 'TAFJUser' to file 'C:\Temenos\3rdParty\AS\jboss-eap-7.2\standalone\configuration\application-users.properties'
Added user 'TAFJUser' to file 'C:\Temenos\3rdParty\AS\jboss-eap-7.2\domain\configuration\application-users.properties'
Added user 'TAFJUser' with groups TAFJAdmin to file 'C:\Temenos\3rdParty\AS\jboss-eap-7.2\standalone\configuration\application-roles.properties'
Added user 'TAFJUser' with groups TAFJAdmin to file 'C:\Temenos\3rdParty\AS\jboss-eap-7.2\domain\configuration\application-roles.properties'
Is this new user going to be used for one AS process to connect to another AS process?
e.g. for a slave host controller connecting to the master or for a Remoting connection for server to server EJB calls.
yes/no? no
Press any key to continue . . .

C:\Temenos\3rdParty\AS\jboss-eap-7.2\bin>
```

4. Use URL `http://localhost:9089/TAFJEE` to access the TAFJEE servlet.

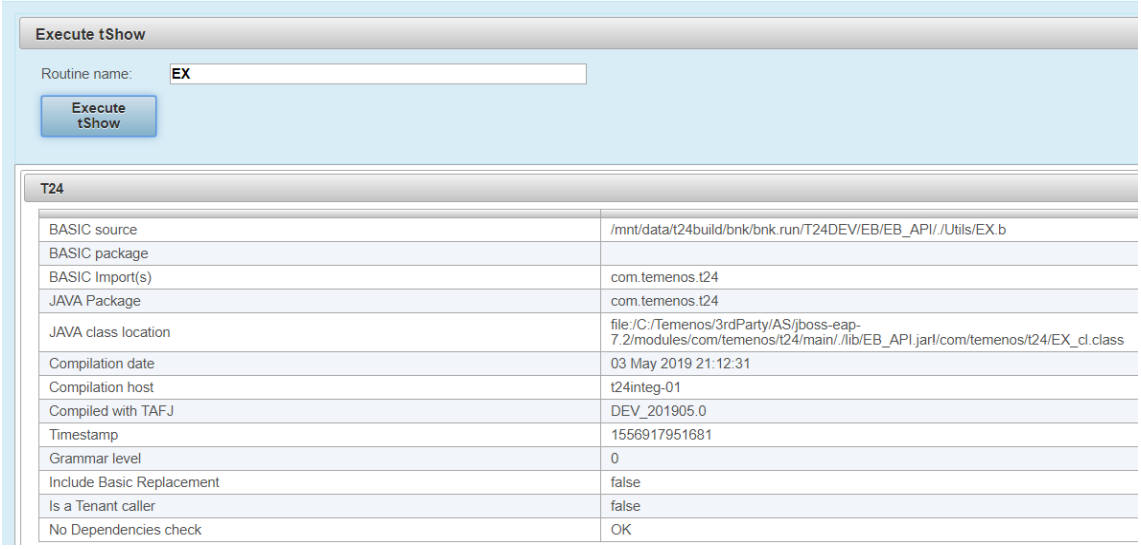


5. Enter the application user credentials. This gives you access to all the functions in TAFJEE.

   Note that some of the tools are protected and require TAFJ user credentels.



6. Click tShow servlet to view the compilation details of a Transact routine on the application server.

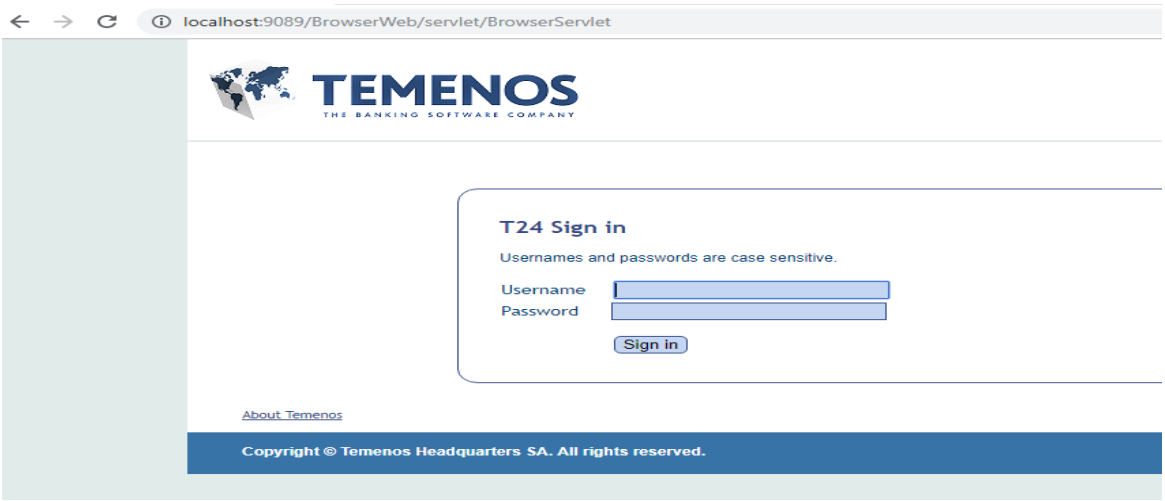


BASIC source	/mnt/data/t24build/bnk/bnk.run/T24DEV/EB/EB_API/./Utils/EX.b
BASIC package	
BASIC Import(s)	com.temenos.t24
JAVA Package	com.temenos.t24
JAVA class location	file:/C:/Temenos/3rdParty/AS/jboss-eap-7.2/modules/com/temenos/t24/main/./lib/EB_API.jar!/com/temenos/t24/EX_cl.class
Compilation date	03 May 2019 21:12:31
Compilation host	t24integ-01
Compiled with TAFJ	DEV_201905.0
Timestamp	1556917951681
Grammar level	0
Include Basic Replacement	false
Is a Tenant caller	false
No Dependencies check	OK

13.3 Accessing BrowserWeb

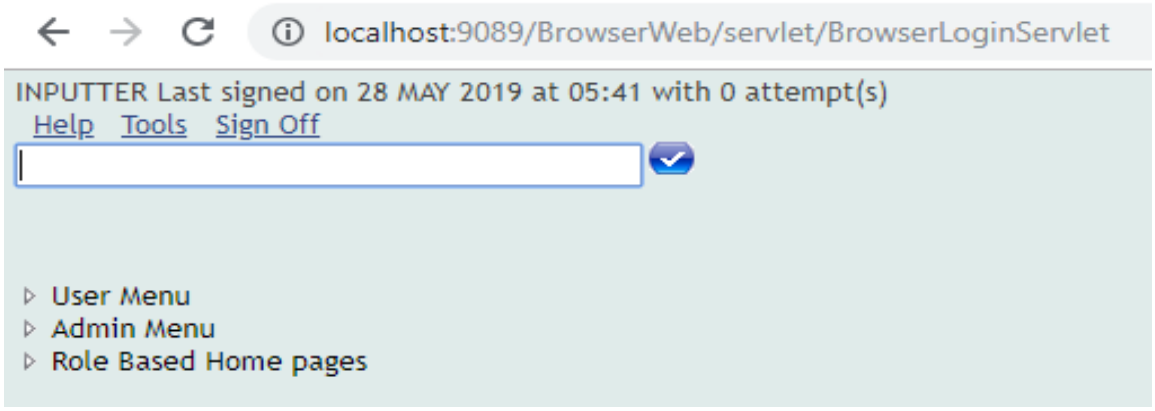
Procedure

1. Access BrowserWeb using URL: `http://localhost:9089/BrowserWeb`.

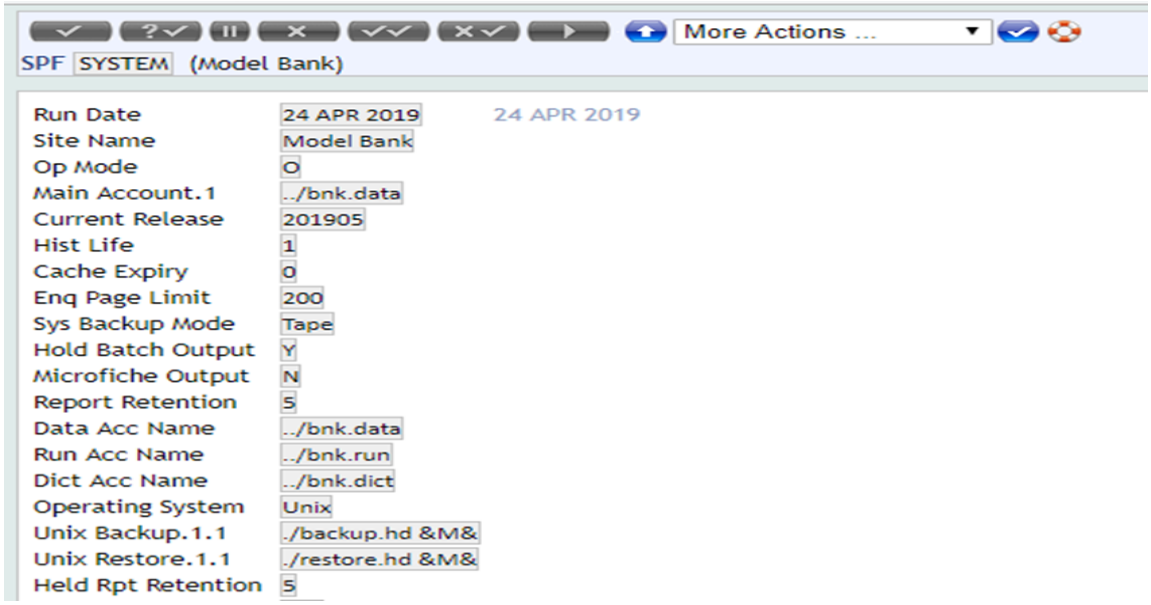


2. Login to Browser using a valid Transact user name and password .

   The landing page is now displayed.



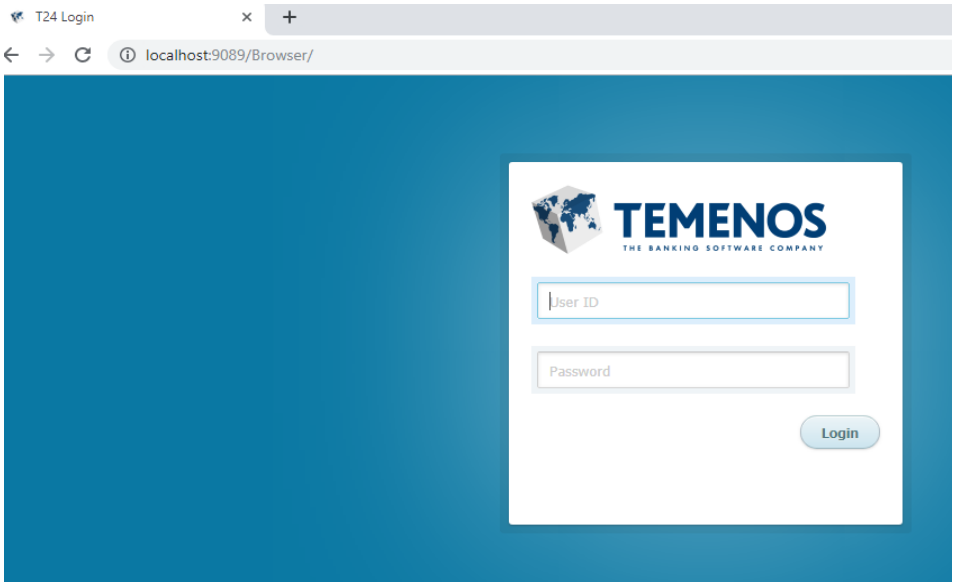
3. Launch any Transact application (e.g.SPF).



13.4 Accessing UXP Browser

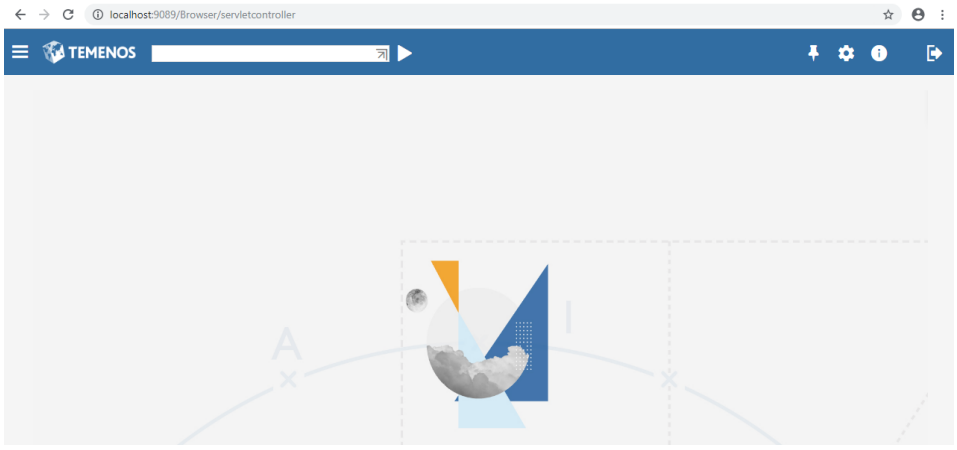
Procedure

1. Use URL: http://localhost:9089/Browser to access the browser.

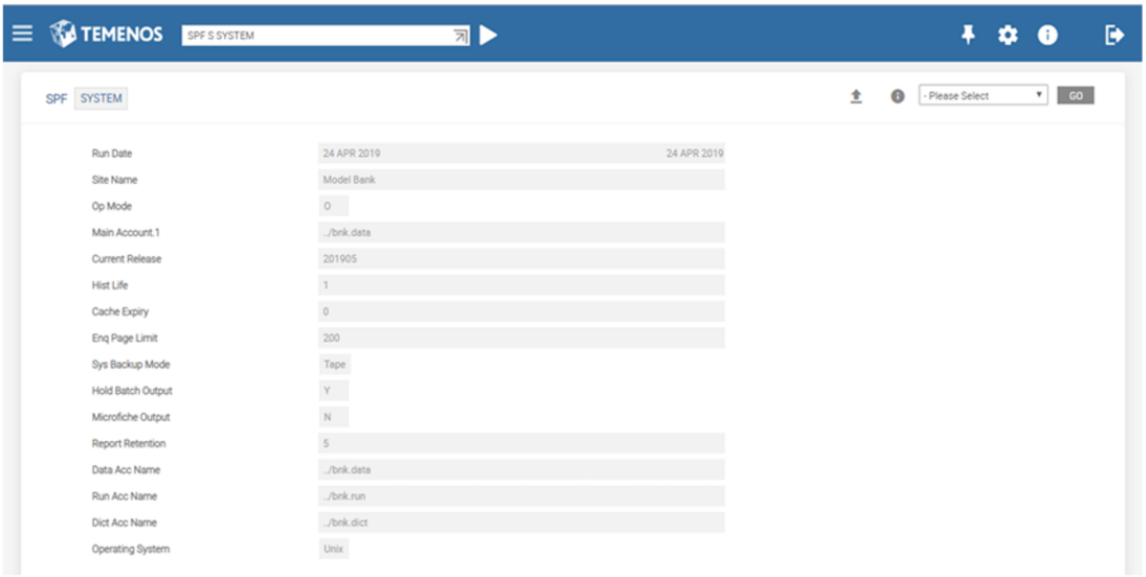


2. Login to Browser using a valid Transact Username and Password.

   The landing page is now displayed.



3. Launch any Transact application (e.g.SPF).

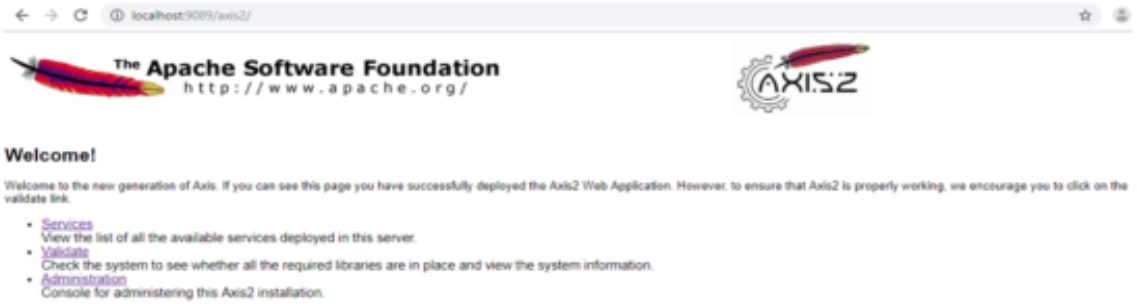


13.5 Using the Axis2 servlet

13.5.1 Accessing the Axis2 servlet

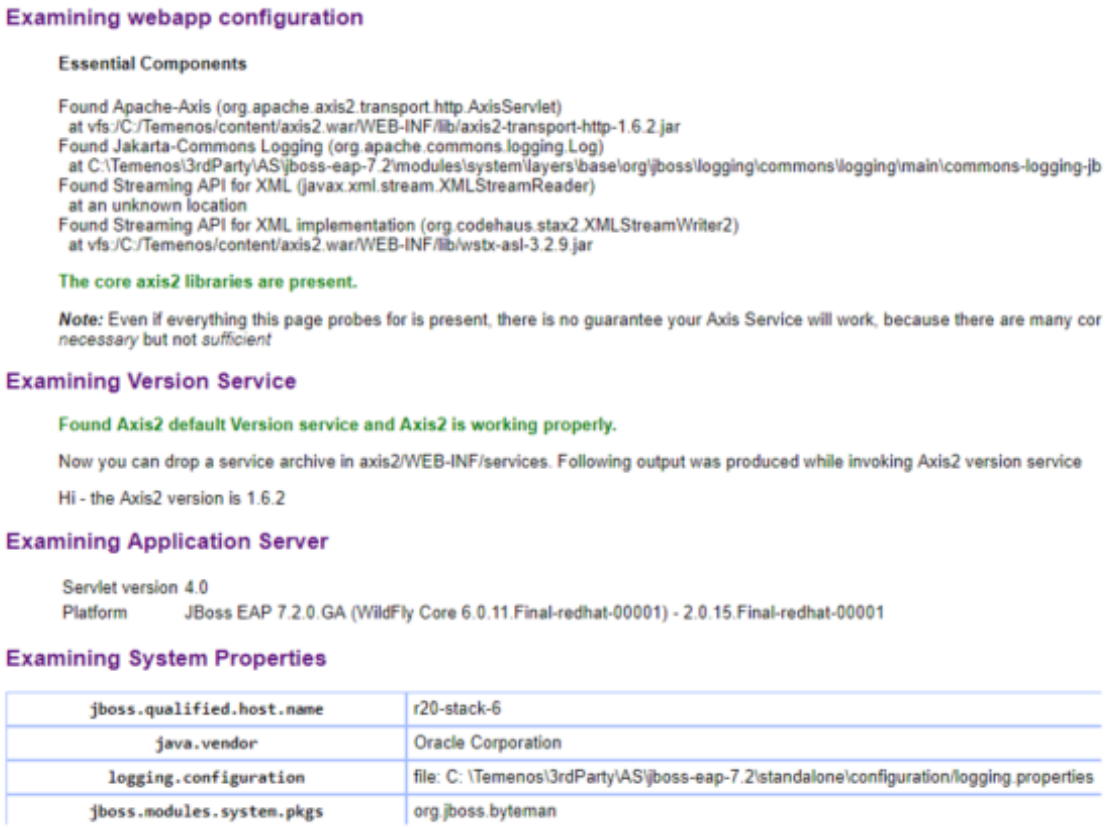
Procedure

1. Use URL `http://localhost:9089/axis2` to access the servlet.



2. Select **Validate**.

The display will indicate whether the web services are working correctly.



Examining webapp configuration

Essential Components

Found Apache-Axis (org.apache.axis2.transport.http.AxisServlet)
at vfs:/C:/Temenos/content/axis2.war/WEB-INF/lib/axis2-transport-http-1.6.2.jar
Found Jakarta-Commons Logging (org.apache.commons.logging.Log)
at C:\Temenos\3rdParty\AS\jboss-eap-7.2\modules\system\layers\base\org\jboss\logging\commons\logging\main\commons-logging-jb
Found Streaming API for XML (javax.xml.stream.XMLStreamReader)
at an unknown location
Found Streaming API for XML implementation (org.codehaus.stax2.XMLStreamWriter2)
at vfs:/C:/Temenos/content/axis2.war/WEB-INF/lib/wstx-asl-3.2.9.jar

The core axis2 libraries are present.

Note: Even if everything this page probes for is present, there is no guarantee your Axis Service will work, because there are many cor
necessary but not *sufficient*

Examining Version Service

Found Axis2 default Version service and Axis2 is working properly.

Now you can drop a service archive in axis2/WEB-INF/services. Following output was produced while invoking Axis2 version service

Hi - the Axis2 version is 1.6.2

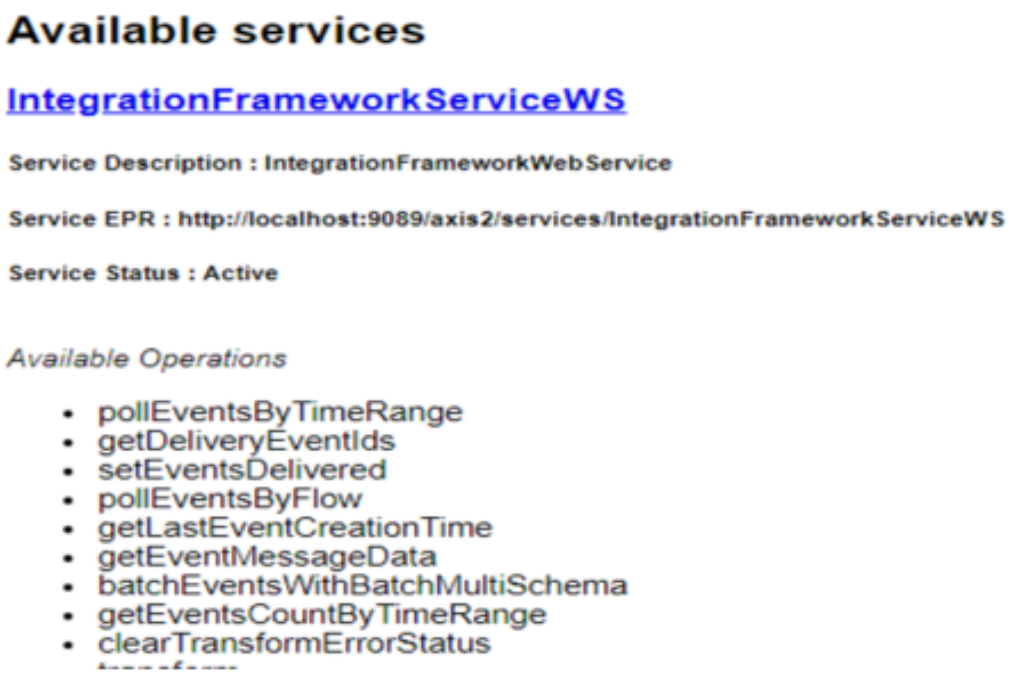
Examining Application Server

Servlet version 4.0
Platform JBoss EAP 7.2.0.GA (WildFly Core 6.0.11.Final-redhat-00001) - 2.0.15.Final-redhat-00001

Examining System Properties

jboss.qualified.host.name	r20-stack-6
java.vendor	Oracle Corporation
logging.configuration	file: C: \Temenos\3rdParty\AS\jboss-eap-7.2\standalone\configuration/logging.properties
jboss.modules.system.pkgs	org.jboss.byteman

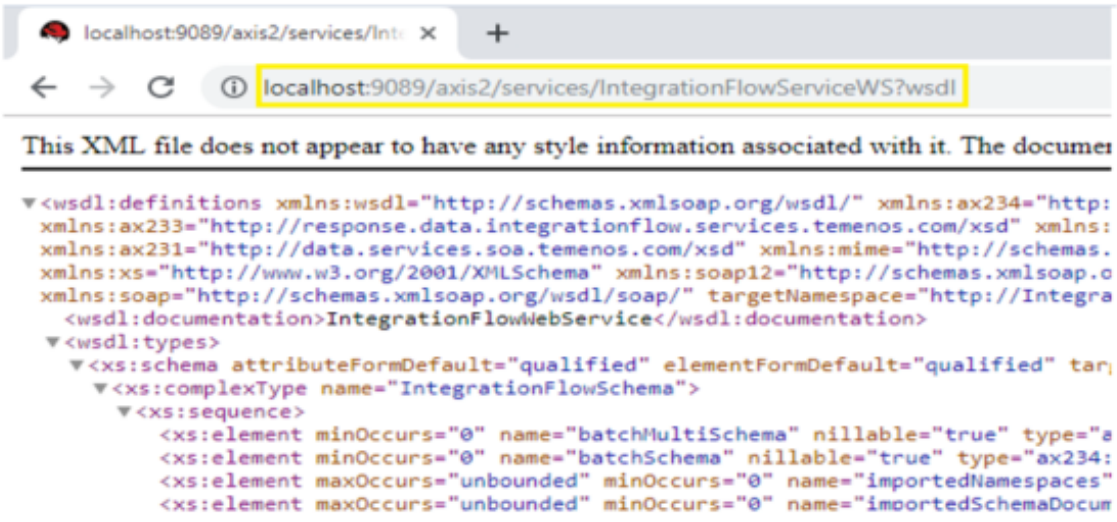
3. Select **Services**. The services deployed in Axis2 are displayed.



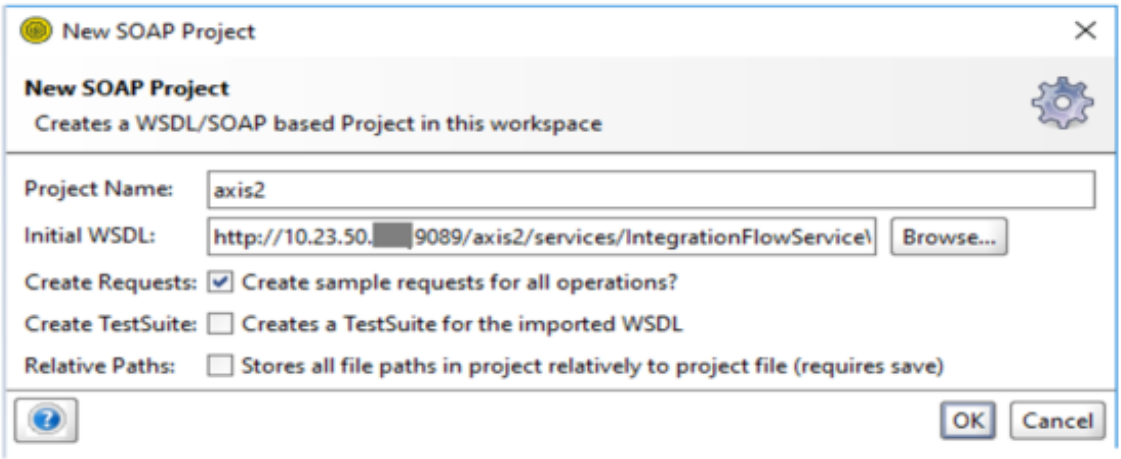
13.5.2 Integrated testing of Axis2 and Transact

Procedure

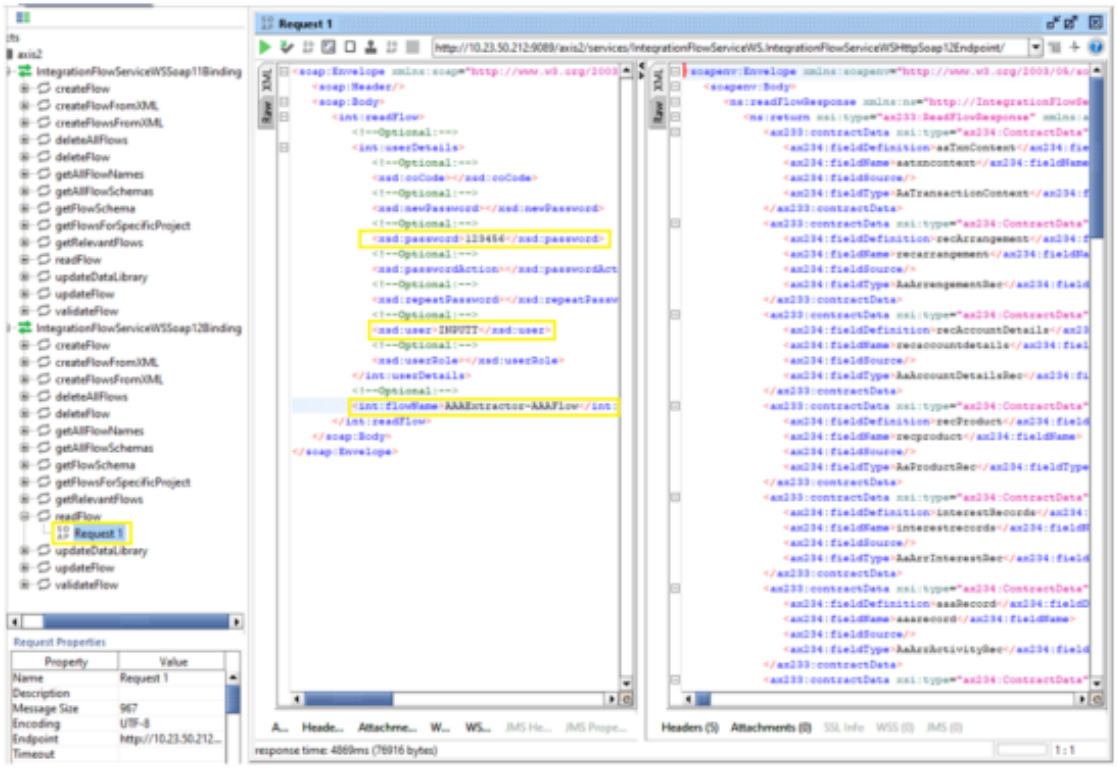
1. In the Axis2 servlet, select **Services**.
2. Select a service to display the wsdl (the example illustrated below is for IntegrationFlowService).



3. Using a third party tool (such as SOAPUI), create a new SOAP project using wsdl.



4. Create a new read flow. Pass the parameters as illustrated below.



If SOAP responds correctly, it indicates that the selected service can communicated correctly with Transact.

This completes deployment of Transact on TAFJ on JBoss-EAP-7.2 with SQL Server 2017 database. You will now be able to perform business operations in Transact using BrowserWeb and UXP Browser.