Contents

Configuring the WebLogic Tuxedo Connector ........................................... 1-1

  Build the Simpapp Example .......................................................... 1-2
  Create a WTC Service ................................................................. 1-3
  Create a Local Tuxedo Access Point .............................................. 1-3
  Create a Remote Tuxedo Access Point .......................................... 1-4
  Create Exported Services ......................................................... 1-4
  Create Imported Services .......................................................... 1-5
  Target mySimpapp to the examplesServer ................................... 1-5
  Register TDOM1 as a WebLogic Server User .................................. 1-5

Configuring Tuxedo ............................................................................ 1-6

Run the Example .............................................................................. 1-7

  WebLogic Server to Tuxedo Interoperability ................................ 1-7
  Tuxedo to WebLogic Server Interoperability ............................... 1-8
WebLogic Tuxedo Connector Quick Start Guide

Note: For more detailed information on how to configure the WebLogic Tuxedo Connector for this WebLogic Server Release, see the WebLogic Tuxedo Connector Programmer's Guide at http://e-docs.bea.com/wls/docs81/wtc_admin/index.html.

The following sections describe how to use the WebLogic Server console to configure WebLogic Tuxedo Connector to allow WebLogic Server to interoperate with Tuxedo Releases 6.5 and higher:

- Configuring the WebLogic Tuxedo Connector
- Configuring Tuxedo
- Run the Example

Configuring the WebLogic Tuxedo Connector

Note: This section summarizes how to configure the WebLogic Tuxedo Connector on a Windows platform. UNIX users can adapt the instructions by making appropriate substitutions such as replacing the “\” with “/” and “.cmd” with “.sh”.

This example extends the Tuxedo simpapp application to run over Tuxedo Domains (TDomains). This allows clients of the TOUPPER service to run on either the Tuxedo server or the WebLogic Server examplesServer. The example provides the following services:

- **TOUPPER**: A Tuxedo service that converts a string to upper case. A WebLogic Server client invokes the TOUPPER EJB and connects to the Tuxedo TOUPPER service.
Tolower: A service implemented by an EJB in WebLogic Server. The client for the Tolower service runs on Tuxedo.

The following sections describe how to configure WebLogic Tuxedo Connector using the Administration Console:

- Build the Simpapp Example
- Create a WTC Service
- Create a Local Tuxedo Access Point
- Create a Remote Tuxedo Access Point
- Create Exported Services
- Create Imported Services
- Target mySimpapp to the examplesServer
- Register TDOM1 as a WebLogic Server User

**Build the Simpapp Example**

**Note:** You may want to enable tracing to monitor WebLogic Tuxedo Connector. See Monitoring the WebLogic Tuxedo Connector located at http://e-docs.bea.com/wls/docs81/wtc_admin/troubleshooting.html.

Use the following steps to build the simpapp example:

1. Boot your WebLogic examplesServer.
2. Open a new shell window and set environment variables using the SAMPLES_HOME\domains\examples\setExamplesEnv.cmd file.
3. Change directories to your SAMPLES_HOME\server\examples\src\examples\wtc\atmi\simpapp directory.
4. Build the wtc_toupper.jar file using ant. This will deploy the EJB on WebLogic Server. Enter the following command: ant
5. Change directories to the SAMPLES_HOME\server\examples\src\examples\wtc\atmi\simpapp directory.
6. Build the wtc_tolower.jar file. This will deploy the EJB on WebLogic Server. Enter the following command: ant
7. Launch the Administration Console in your browser. Use the following URL: http://your_machine:7001/console. Replace your_machine with the IP address for your machine or your machine name.

8. In the navigation tree, Click Deployments→EJB Modules and confirm that the wtc_tolower.jar and wtc_toupper.jar are deployed.

Create a WTC Service

Use the following steps to create and configure a WTC service using the WebLogic Server Console:

1. Click Services in the navigation tree.
2. Right-click WTC.
3. Click Configure a New WTC Service.
4. Enter the Name of your WTC service in the Name field. Example: mySimpapp
5. Click Create.
6. Your new WTC Service appears in the navigation tree.

Create a Local Tuxedo Access Point

Note: When configuring the Network Address for a local access point, the port number used should be different from any port numbers assigned to other processes. Example: Setting the Network Address to //mymachine:7001 is not valid if the WebLogic Server listening port is assigned to //mymachine:7001.

Use the following steps to configure a local Tuxedo access point:

1. Click mySimpapp in the navigation tree.
2. Click the Contents tab.
3. Select Local APs and click Configure a new Local Tuxedo Access Point.
4. Enter the following values for the following fields on the General tab:
   - Access Point: myLocalAp
   - AccessPoint ID: TDOM2
   - Network Address: the network address and port of your local access point
Example: //123.123.123.123:5678

5. Click Create.

6. If you are connecting to a Tuxedo 6.5 domain, do the following:
   a. Click the Connections tab.
   b. Set the Interoperate field to Yes.
   c. Click Apply.

Create a Remote Tuxedo Access Point

Use the following steps to configure a remote Tuxedo access point:

1. Click mySimpapp in the navigation tree.

2. Click the Contents tab.


4. Enter the following values for the following fields on the General tab:
   - Access Point: myRemoteAP
   - AccessPoint ID: TDOM1
   - Local Access Point: myLocalAp
   - Listen Address: the network address and port of your remote access point
     Example: //123.123.123.123:1234

5. Click Create.

Create Exported Services

Use the following steps to configure an exported service:

1. Click mySimpapp in the navigation tree.

2. Click the Contents tab.

3. Select Exported and click Configure a new Exported Service

4. Enter the following values for the following fields on the General tab:
Configuring the WebLogic Tuxedo Connector

Resource Name: TOLOWER
Local Access Point: myLocalAp
EJB Name: tuxedo.services.TOLOWERHome
Remote Name: TOLOWER

5. Click Create.

Create Imported Services
Use the following steps to configure an imported service:

1. Click mySimpapp in the navigation tree.
2. Click the Contents tab.
3. Select Imported and click Configure a new Imported Service
4. Enter the following values for the following fields on the General tab:
   Resource Name: TOUPPER
   Local Access Point: myLocalAp
   Remote Access Point List: myRemoteAP
   Remote Name: TOUPPER

5. Click Create.

Target mySimpapp to the examplesServer

1. Click mySimpapp in the navigation tree.
2. Click the Target and Deploy tab.
3. Click the checkbox for the examplesServer.
4. Click Apply

Register TDOM1 as a WebLogic Server User
Use the following steps to register TDOM1 as a WebLogic Server user:

1. Click Security in the navigation tree.
2. Click Realms.
3. Click myRealm.
4. Click on Users.
5. Click the Configure a new User.
6. In the General tab, do the following:
   a. Add TDOM1 in the Name field.
   b. Enter and validate a password.
   c. Click apply.

**Configuring Tuxedo**

Use the following steps to configure your Tuxedo domain:

1. Your PATH environment variable needs to include the path of your C compiler. Use set PATH to check the status and add the path if necessary.
2. Copy the simpapp example from your Tuxedo installation and create a working Tuxedo simpapp directory.
3. Change directories to your working Tuxedo simpapp directory.
4. Set environment variables using the setEnv.cmd located at TUXDIR. Update the following parameters:
   - TUXDIR - base directory of the TUXEDO Software
   - APPDIR - base directory of the sample program
5. Build the clients:
   - buildclient -o simpcl -f simpcl.c
   - buildserver -o simpserv -f simpserv.c -s TOUPPER
6. Copy the ubbdomain and dom1config files from the SAMPLES_HOME\server\examples\src\examples\wtc\atmi\simpapp directory to your Tuxedo simpapp directory.
7. Copy the tolower.c file from the SAMPLES_HOME\server\examples\src\examples\wtc\atmi\simpserver directory to your Tuxedo simpapp directory.
8. Modify the ubbdomain for your Tuxedo environment. This includes setting the pathnames for APPDIR, TUXCONFIG, and TUXDIR and setting the machine name. Replace all <braced> items with information for your environment.

Example:

```plaintext
APPDIR="\home\me\simpapp"
TUXCONFIG="\home\me\simpapp\tuxconfig"
TUXDIR="\usr\tuxedo"
```

9. Load the ubbdomain file: `tmloadcf -y ubbdomain`

10. Modify the dom1config for your Tuxedo environment. This includes creating log devices and updating the network addresses.

Example:

```plaintext
DMTLOGDEV="d:\my_apps\tlog"
AUDITLOG="d:\my_apps\aud"
TDOM1 NWADDR="/TuxedoMachine:1234"
TDOM2 NWADDR="/WTCMachine:5678"
```

11. Load the dom1config file.

```plaintext
set BDMCONFIG=d:\mydomain\simpapp\bdmconfig
dmloadcf -y dom1config
```

12. Build the tolower client

```plaintext
buildclient -f tolower.c -o tolower
```

13. Boot the Tuxedo domain

```plaintext
tmboot -y
```

**Run the Example**

Run each client to demonstrate interoperability between Tuxedo and WebLogic Server.

**WebLogic Server to Tuxedo Interoperability**

Start a WebLogic Server client from the `SAMPLES_HOME\server\examples\src\examples\wtc\atmi\simpapp` directory to invoke the TOUPPER EJB and connect to the Tuxedo TOUPPER service. Use the following command:

```plaintext
ant run
```
The Tuxedo service replies to your WebLogic Server application with:

```java
Buildfile: build.xml
run:
[java]
[java] Beginning statefulSession.Client...
[java]
[java] Creating Toupper
[java]
[java] converting allcaps
[java] converted to: ALLCAPS
[java]
[java] End statefulSession.Client...
[java]
BUILD SUCCESSFUL
```

**Tuxedo to WebLogic Server Interoperability**

Run the `tolower` client from the Tuxedo `simpapp` directory to invoke the Tolower EJB and return the results to the client. Use the following command:

```
tolower ALLSMALL
```

The WebLogic Server service replies to your Tuxedo client with:

```
Returned string is: allsmall
```