

ORIGINATOR: TAXUD/R5	ISSUE DATE: 18/11/2010	VERSION: 5.00-EN
<p style="text-align: center;">SUBJECT:</p> <p style="text-align: center;">TARIC3 Installation Procedure Manual</p>		

DOCUMENT HISTORY

EDI.	REV.	DATE	DESCRIPTION	ACTION	PAGES
0	01	25/03/2009	Initial version	Insert	All
0	10	04/06/2009	SFR	Update	As required
1	00	06/07/2009	Implementation of review comments. Sent for acceptance.	Update	As required
1	01	03/09/2009	Update of build variables and new runtime variable.	Insert, Update	Sections 4.2 and 4.5.1
1	10	09/09/2009	Sent for Review.	-	-
2	00	30/09/2009	Implementation of review comments. Update of database conversion procedure. Sent for Acceptance.	Update	Section 4.11
2	01	30/10/2009	Minor updates.	Update Insert	Section 4.5.1, 4.9 and 5.5.1.3 4.12
2	10	16/04/2010	Sent for Review.	-	-
3	00	18/05/2010	Sent for Acceptance.	Update	As requested
3	10	26/05/2010	The configuration of the TimerQueue has been updated.	Update	5.5.1.4
3	20	08/06/2010	Sent for Review.	-	-
4	00	22/06/2010	Sent for Acceptance.	-	-

TAXUD/R5 – TARIC3 Installation Procedure Manual	
DOCUMENT HISTORY	Ref: TARIC3-IPR

4	01	19/10/2010	Removed QUOTA2 references	Update	Sections 4.6, 4.9 and 5
4	02	21/20/2010	Internal Review.	-	-
4	10	29/10/2010	Sent for Review.	-	-
5	00	18/11/2010	Sent for Acceptance.	Update	As requested

TAXUD/R5 – TARIC3 Installation Procedure Manual	
TABLE OF CONTENTS	Ref: TARIC3-IPR

TABLE OF CONTENTS

1	INTRODUCTION	6
1.1	Objective of this document	6
1.2	Intended audience.....	6
1.3	Structure of this document	6
1.4	Abbreviations and acronyms.....	6
1.5	Reference documents	7
1.6	Applicable documents	7
2	OVERVIEW	8
3	INSTALLATION PREREQUISITES	9
3.1	WebLogic environment.....	9
3.2	Java environment.....	9
3.3	Persistent storage	10
3.4	Shell environment.....	10
3.5	UM	10
3.6	Runtime properties information	10
4	INSTALLATION AND CONFIGURATION	11
4.1	Extraction of the sources	11
4.2	Creation and customisation of the build configuration file.....	11
4.3	Building the runtime environment.....	12
4.4	Extraction of the runtime archive	12
4.5	Configuration of the runtime environment.....	13
4.5.1	Application runtime environment	13
4.5.2	Group runtime environment	13
4.6	Creation of the WebLogic resources	14
4.7	Installation of the database schema.....	16
4.8	Security management	16
4.9	Deployment of the application in WebLogic	16
4.10	Configuration of the application	18
4.11	Conversion Tools	18
4.11.1	Prerequisite	18
4.11.2	Customization	19
4.11.3	Conversion	19
4.12	Setup of TARIC2 compatible reference database	20

TAXUD/R5 – TARIC3 Installation Procedure Manual	
TABLE OF CONTENTS	Ref: TARIC3-IPR

4.13	Setup of environment.name in the TARIC3 GUI JNLP file.....	21
5	DETAILS OF THE WEBLOGIC RESOURCES	22
5.1	Machine	22
5.2	Server	22
5.2.1	Work managers	22
5.3	Persistent stores	22
5.4	JMS servers.....	22
5.5	JMS Modules.....	23
5.5.1	TARIC3 JMS module.....	23
5.6	JDBC data sources.....	27
5.6.1	XA data sources	27
5.6.2	Non-XA Data sources.....	28
5.7	TARIC3 server JNDI tree validation	29
6	DIRECTORY CONTENT	32

TAXUD/R5 – TARIC3 Installation Procedure Manual	
TABLE OF TABLES	Ref: TARIC3-IPR

TABLE OF TABLES

Table 1. TARIC3 WebLogic resources	15
Table 2. TARIC3 J2EE Modules.....	18
Table 3. Required JMS queues for the TARIC3JMSServer	25
Table 4. TARIC3 JMS queues properties.....	25
Table 5. TARIC3 JMS queues properties for Timer Queue	26
Table 6. Required JMS topics for the TARIC3JMSServer.....	26
Table 7. TARIC3 JMS topic properties.....	26
Table 8. TARIC3 JMS topic properties for Timer Topic	26
Table 9. TARIC3 XA data source	28
Table 10. TARIC3 non-XA data source.....	29
Table 11. TARIC3 JNDI tree.....	31
Table 12. Content of an application group hierarchy.....	32
Table 13. Content of an application directory hierarchy	32

TAXUD/R5 – TARIC3 Installation Procedure Manual	
INTRODUCTION	Ref: TARIC3-IPR
OBJECTIVE OF THIS DOCUMENT	

1 INTRODUCTION

1.1 Objective of this document

This document describes the installation procedure for the TARIC3 application. The basics of the installation procedure are shared by all new customs applications of the Directorate General Taxation and Customs Union (DG TAXUD).

1.2 Intended audience

This document is intended for people responsible for the installation and the configuration of the different customs applications at DG-TAXUD that use the BEA WebLogic application server infrastructure with some specific parts for the installation of TARIC3.

1.3 Structure of this document

Chapter 2 contains a general description of the installation procedure. Chapter 3 describes the prerequisites for a successful installation. Chapter 4 describes the installation in more detail whereas chapter 5 describes the resources that must be configured in the BEA WebLogic server. Finally, chapter 6 succinctly describes the runtime directory hierarchy.

1.4 Abbreviations and acronyms

ANT	Another Neat Tool
CCN	Common Communication Network
CSI	Common Service Interface
DBA	DataBase Administrator
DG TAXUD	Directorate General Taxation and Customs Union
EAR	Enterprise ARchive
EJB	Enterprise Java Bean
HTTP	HyperText Transfer Protocol
J2EE	Java 2 Enterprise Edition
JDBC	Java DataBase Connectivity
JEE	Java Enterprise Edition
JDK	Java Development Kit
JMS	Java Messaging Service
JNDI	Java Naming and Directory Interface

TAXUD/R5 – TARIC3 Installation Procedure Manual	
INTRODUCTION	Ref: TARIC3-IPR
REFERENCE DOCUMENTS	

JRE	Java Runtime Environment
MSA	Member State Administration
RAR	Resource adaptor ARchive
T3	WebLogic T3 protocol for Java-to-Java connections
TAR	Tape ARchive
TARIC	TARif Intégré Communautaire
UM	User Management
URL	Uniform Resource Locator
UUID	Universally Unique Identifier
WAR	Web ARchive
WLS	WebLogic Server
XA	X/Open XA standard for distributed transaction processing

1.5 Reference documents

[UM-INS-001]	“UM Installation and Administration Guide”, ref. UM-INS-001, 2.00
[TARIC3-AMN]	“TARIC3 Administration Manual”, ref. TARIC3-AMN, 3.00

1.6 Applicable documents

None.

TAXUD/R5 – TARIC3 Installation Procedure Manual	
OVERVIEW	Ref: TARIC3-IPR

2 OVERVIEW

This chapter contains a general description of the TARIC3 installation procedure. The basic procedure is very similar for all application systems that use the BEA WebLogic Server for the implementation of the customs related applications. The aim of this overview is to provide the reader with a global view of the installation procedure and the tools used.

The BEA WebLogic Server (WLS) has been chosen as the platform to host the new implementations of the DG TAXUD customs applications. WLS (Version 10.3) is an implementation of Java Enterprise Edition 5 (JEE5). The customs applications have as common requirement to keep information in a persistent storage. All persistent data are stored in the existing TARIC2 schema which is inside an Oracle 10 (10.2.0.3) relational database.

The installation of the application itself consists of two distinct steps. First the application is compiled, using configuration parameters specified in a build configuration file (`build.properties`), resulting in the creation of one or more Enterprise Application Archives (with extension “`.ear`”), possibly Web Archives (with extension “`.war`”) and/or Resource Adaptor Archives (with extension “`.rar`”). These archives should then be extracted on the platform hosting the runtime environment (possibly by transferring to a different machine). The extraction must be done in a chosen directory and the application must be configured in this environment. Once configured, the application can be deployed in a managed server, which can be done using the standard WLS console interface.

The build phase of the installation procedure uses the ANT facility to call different `build.xml` in the relevant subdirectories. A shell script (`configure`) is used to perform certain initial steps during configuration and to create the required database schema, or perform upgrades, if required (for future releases). The actual make of the Java components is performed using the ANT facility.

TAXUD/R5 – TARIC3 Installation Procedure Manual	
INSTALLATION PREREQUISITES	Ref: TARIC3-IPR
WEBLOGIC ENVIRONMENT	

3 INSTALLATION PREREQUISITES

This chapter details the prerequisites for a successful installation of the application.

3.1 WebLogic environment

The application uses the BEA WebLogic application server release 10.3. It might not work properly with later WebLogic application server releases.

The JRE 1.6.0_5 (the JDK can be used instead) must be used to run the WLS. It is installed with WebLogic. It can also be downloaded from the Sun internet site.

See [Java 6.0 update 5](#)

Note that with the WebLogic 10.3 server, the JEE classes required to compile the application are no longer centralized in one JAR file. This JAR file (wlfullclient.jar) can be created by following the instructions given by BEA:

<http://edocs.bea.com/wls/docs103/client/jarbuilder.html#wp1078098>

Section: 'Creating a wlfullclient.jar for JDK 1.6 client applications'.

3.2 Java environment

The following external packages are required and need to be present on the build machine. Nevertheless, no special installation procedure is required as they are included automatically during the build:

Java SDK 6.0 is required for compilation and runtime; the version 1.6.0_5 is included with the installation of the BEA WLS. Note that the graphical user interface is a java swing client, so the users must install the JRE on their individual computer. It can also be downloaded from the Sun internet site.

See [Java 6.0 update 5](#)

Apache Jakarta ANT 1.6.5 is required for application compilation and packaging. This package is included in the delivery. It can also be downloaded from the Apache internet site.

See <http://archive.apache.org/dist/ant/binaries/>

The following external packages are used for building and running the server and included in the delivery. They are included automatically during the build:

Xerces 2.9.1 is required for compilation and at runtime.

See <http://archive.apache.org/dist/xml/xerces-j/>

Xalan 2.7.1 is required for compilation and at runtime.

See <http://archive.apache.org/dist/xml/xalan-j/>

TAXUD/R5 – TARIC3 Installation Procedure Manual	
INSTALLATION PREREQUISITES	Ref: TARIC3-IPR
PERSISTENT STORAGE	

3.3 Persistent storage

Nearly all persistent data for the application are stored in the Oracle 10 (10.2.0.3) relational database. The installation procedure assumes that the database has been properly installed and that a login is available with the proper grants and table-spaces to create the new database objects. Therefore the local DBA should create the proper user and the table-space prior to the deployment of the application and grant the user the proper roles and accesses.

The following roles must be granted to the login:

- CREATE SESSION;
- CREATE TABLE;
- CREATE VIEW;
- CREATE SYNONYM;
- CREATE SEQUENCE;
- CREATE TRIGGER;
- CREATE PROCEDURE;
- CREATE CLUSTER.

In addition, the user must have select access to the DBA_PENDING_TRANSACTIONS view. Creation of the database schema itself is performed as part of the installation.

3.4 Shell environment

There are no external dependencies on environmental variables to install the application. All such variables are initialised from specific configuration files that are edited during either the installation or deployment phase of the system.

3.5 UM

UM must be installed and running, to be able to create groups and users for the application.

3.6 Runtime properties information

You must have the information needed to complete the runtime.properties (see §4.5).

TAXUD/R5 – TARIC3 Installation Procedure Manual	
INSTALLATION AND CONFIGURATION	Ref: TARIC3-IPR
EXTRACTION OF THE SOURCES	

4 INSTALLATION AND CONFIGURATION

This chapter contains a description of the actual installation procedure. A full installation starting from the sources consists of the following distinct steps:

1. Create the installation hierarchy by extraction of the delivered source archive;
2. Create and customise the build configuration file in the source hierarchy;
3. Build a runtime environment from the source hierarchy, and possibly transfer it to the runtime platform (if different from the build platform);
4. Extract the runtime archive in the runtime deployment directory hierarchy;
5. Configure the runtime environment;
6. Create the necessary WebLogic resources;
7. Install (or upgrade) the database schema if required;
8. Setup the security of the application (groups and users);
9. Deploy the application in the WLS;
10. Configure the application;
11. Execute the conversion tools.

The build procedure does not depend on the current environmental variable settings.

4.1 Extraction of the sources

Extract the source archive in a directory of choice, further referred to as <SRCDIR>, e.g. using the following commands:

```
$ cd <SRCDIR>
$ tar -xvf <archive_name>
```

The actual name of the source archive is given in the release notes. It is of the format <name>_<release_number>.src.tar.

4.2 Creation and customisation of the build configuration file

Edit the `build.properties` file to reflect the local configuration, most likely only the values for the following variables need to be changed:

WL_HOME: the absolute path of the directory in which the BEA WebLogic Software has been installed, it is the directory containing the “server” directory amongst others.

WLS_HOME: the absolute path of the BEA WebLogic Software “server” directory.

JAVA_HOME: the absolute path of the JDK hierarchy (version 1.6.0_update5).

TAXUD/R5 – TARIC3 Installation Procedure Manual	
INSTALLATION AND CONFIGURATION	Ref: TARIC3-IPR
BUILDING THE RUNTIME ENVIRONMENT	

The following variable is used to locate the XEP license file at build time:

xep.license.file: path to the XEP license file (license.xml since XEP 4.10). This file will be copied to the XEP configuration embodied in the TARIC3 EAR.

The remaining variables should normally not be changed.

4.3 Building the runtime environment

The next command calls the top-level `build.xml` ANT script to build the different archives required for running the application:

```
$ ./ant.sh build
```

If errors occur, the build aborts with an error message. Normally, this should not occur; the main reason will probably be an error in a parameter of the build properties. If necessary, re-edit the `build.properties` file and restart the build with `ant.sh build`. To save the output of the operation in a log file, pipe the output of the above command in `tee` as in:

```
$ ./ant.sh build 2>&1 | tee [-a] logfile
```

The `-a` option appends log information to an existing log file. To clean up all targets (for instance for a complete recompilation), use the following command:

```
$ ./ant.sh clean
```

Once the build completes, a deployment archive `TARIC3_<release_number>.run.tar` is present in the `<SRCDIR>` directory. This archive will contain the following components:

- a runtime configuration file and configure script to configure the runtime environment;
- one or more enterprise application archives (“`.ear`”);
- zero or more web application archives (“`.war`”).

The output directory and the name of the archive are specified in the `build.properties` file, the default name is `<application_name>_<release_number>.run.tar`, for instance `TARIC3_1.0.0.run.tar`. The runtime archive created must be copied to the target machine. Then, it must be extracted in a chosen directory referenced in the rest of the document as the `<GROUP>` directory.

4.4 Extraction of the runtime archive

As described for the build step, make sure that the proper UNIX, BEA WLS, and Oracle components are properly installed and configured.

Extract the runtime archive (referred to as `<runtime_archive>`) in the `<GROUP>` directory on the deployment platform. This directory should be empty if it is the first application to be installed in that group.

The following command can be used:

```
$ cd <GROUP>
$ tar -xvf <runtime_archive>
```

TAXUD/R5 – TARIC3 Installation Procedure Manual	
INSTALLATION AND CONFIGURATION	Ref: TARIC3-IPR
CONFIGURATION OF THE RUNTIME ENVIRONMENT	

This extraction creates the application-specific subdirectory `<GROUP>/app/taric3` which contains most of the TARIC3 specific runtime files.

4.5 Configuration of the runtime environment

4.5.1 Application runtime environment

The TARIC3 subdirectory `<GROUP>/app/taric3` contains a `runtime.properties` file with the TARIC3 specific runtime configuration parameters. Edit this file to reflect the local configuration as the generated copy contains the parameters used in the build environment. Most likely only the values for the following variables need to be changed:

USER: the Oracle user name to create/upgrade the TARIC3 objects in the database schema, this user name should have been created by a DBA and should have been granted privileges given in section 3.3;

PASS: the associated password;

SCHEMA: the name of the schema (same as value for USER);

DB_CONN_URL: the JDBC connection URL pointing to the above schema in the format `jdbc:oracle:thin:<host>:<port>:<instance_name>` (e.g. *[Removed]*);

APP_SERVER: the name of the WLS hosting the application;

APP_SERVER_PORT: the port number on which the WLS hosting the application is listening to;

APP_HOST: the hostname of the server on which the WLS hosting the application is installed;

MACHINE_NAME : the machine name (this is the machine to which the WLS hosting the application will be assigned);

UM_DIR: The full path to the runtime environment of the User Management application. (e.g. `/home/wlsapp/astaxud/groups/group1/app/um`).

log4j.appender.R.File: The full path to the input bridge log file. (e.g. `/home/wls10/groups/group10/log/weblogic-inb.log`).

Next, configure the runtime environment with the command (in `<GROUP>/app/TARIC3` directory):

```
$ ./configure env
```

This command checks the values of some configuration parameters and reports detected errors. If necessary, correct the errors and re-execute the command.

4.5.2 Group runtime environment

The command `(./configure env)` verifies if a `group.properties` file exists in the `<GROUP>` directory. If not, that file is created from a template and has to be edited to

TAXUD/R5 – TARIC3 Installation Procedure Manual	
INSTALLATION AND CONFIGURATION	Ref: TARIC3-IPR
CREATION OF THE WEBLOGIC RESOURCES	

reflect the local configuration. This `group.properties` file will then be common for any other application installed in that group hierarchy.

Most likely only the values for the following variables need to be changed:

BEA_HOME: the directory in which the WebLogic software is installed (e.g. `/opt/bea10`);

JAVA_HOME: the absolute path of the JDK hierarchy;

WL_HOME: the absolute path of the directory in which the BEA WebLogic Software has been installed, it is the directory containing the “server” directory amongst others;

ADMIN_USER: the name of the administrator user of the installed BEA WebLogic Domain, e.g. “*[Removed]*”;

ADMIN_PASS: the password for the administrator user;

ADMIN_T3_URL: the T3 URL of the administrative WLS;

DOM_NAME: the name chosen for the WebLogic Domain.

Re-execute the command (`./configure env`) to achieve the configuration of the application environment. This command checks the values of some configuration parameters and reports detected errors. If necessary, correct the errors and re-execute the command.

4.6 Creation of the WebLogic resources

The machine, the server hosting the TARIC3 application and all the required WebLogic resources must be created. To be able to do this, the WebLogic administration server must be running. The configuration is done with the command:

```
$ ./configure server create
```

If a particular resource already exists, a notification is raised, and the resource is not re-created.

Because some modifications are made in the administrative server of the WebLogic domain, it is advised to re-start the administrative server at this point. This ensures that all newly created or modified resources are available in the different managed servers.

The WebLogic resources are given in the table below.

Detailed information about all resources is given in chapter 5.

Resource name	Resource type
<APP_HOST> ¹	Machine

¹ See 4.5.1

TAXUD/R5 – TARIC3 Installation Procedure Manual	
INSTALLATION AND CONFIGURATION	Ref: TARIC3-IPR
CREATION OF THE WEBLOGIC RESOURCES	

Resource name	Resource type
<APP_SERVER> ²	Server
TARIC3DeadLetterQueue	JMSQueue
TARIC3ErrorQueue	JMSQueue
TARIC3InputQueue	JMSQueue
TARIC3ResultQueue	JMSQueue
TARIC3CcnOutQueue	JMSQueue
TARIC3ReportDeadLetterQueue	JMSQueue
TARIC3ReportInputQueue	JMSQueue
TARIC3ReportResultQueue	JMSQueue
TARIC3ReportErrorQueue	JMSQueue
TARIC3BatchDeadLetterQueue	JMSQueue
TARIC3BatchErrorQueue	JMSQueue
TARIC3BatchInputQueue	JMSQueue
TARIC3BatchResultQueue	JMSQueue
TARIC3BatchTimerQueue	JMSQueue
TARIC3BatchReadyToRunLowQueue	JMSQueue
TARIC3BatchReadyToRunMediumQueue	JMSQueue
TARIC3BatchReadyToRunHighQueue	JMSQueue
TARIC3BatchTimerTopic	JMSTopic
TARIC3BatchJobNotificationTopic	JMSTopic
TARIC3TemporaryTemplate	JMSTemplate
TARIC3JMSServer	JMSServer
TARIC3Store	JMSJDBCStore
TARIC3PagingStore	JMSJDBCStore
TARIC3BatchConnectionFactory	JMSConnectionFactory
TARIC3ReportConnectionFactory	JMSConnectionFactory
TARIC3ConnectionFactory	JMSConnectionFactory
TARIC3ConnectionPool	JDBCConnectionPool
TARIC3NonXAConnectionPool	JDBCConnectionPool
TARIC3NonXADataSource	JDBCTxDataSource
TARIC3ReportDataSource	JDBCTxDataSource
TARIC3DataSource	JDBCDataSource

Table 1. TARIC3 WebLogic resources

If required, the following command will drop all the WebLogic resources except the WebLogic machine and server.

```
$ ./configure server drop
```

² See 4.5.1

TAXUD/R5 – TARIC3 Installation Procedure Manual	
INSTALLATION AND CONFIGURATION	Ref: TARIC3-IPR
INSTALLATION OF THE DATABASE SCHEMA	

4.7 Installation of the database schema

Please refer to the release note to determine the best command to apply to the environment in which the application is installed.

For the initial release, create the new database environment with the following command:

```
$ ./configure database initial
```

As specified for the build step, the output of the operation can be saved in a log file by using in “tee” command.

The `./configure` command without arguments will list the supported upgrades. For instance:

```
current initial/upgrade scripts supported are:
    initial
    rel_1.0.1
    rel_1.1.0
    rel_1.1.1
    ...
```

As an example, to upgrade a database to release 1.1.1, execute the command:

```
$ ./configure database upgrade initial rel_1.1.1
```

4.8 Security management

Configure the security as explained in section “Security management” in the [\[TARIC3-AMN\]](#) to setup the security of the TARIC3 application.

4.9 Deployment of the application in WebLogic

The following enterprise application archive must be deployed in the managed server to be used for the TARIC3 application.

```
s_taric3-<release_number>.ear
```

Please read carefully section “TARIC3 application and module deployment descriptors” in [\[TARIC3-AMN\]](#) to check if any adaptations are needed before deploying the application in WebLogic.

Especially about the following modules:

- g_demo_ejb
- s_taric3_ejb

To be able to do this, the administrative server must be running. The deployment will be done with an exploded distribution of the application archive. To generate that exploded distribution, execute the command:

TAXUD/R5 – TARIC3 Installation Procedure Manual	
INSTALLATION AND CONFIGURATION	Ref: TARIC3-IPR
DEPLOYMENT OF THE APPLICATION IN WEBLOGIC	

```
$ ./configure explode
```

A directory `s_taric3-<release_number>.ear` containing the exploded distribution is created in the exploded directory of the runtime environment. Additionally, during this operation, some default values are configured in the deployment descriptors to take into account some runtime environment parameters (as the location of the TARIC3 temporary directory `<GROUP>/app/taric3/tmp`, etc). Note that the exploded distribution has the advantage of accessing easily the deployment descriptors of the application.

The deployment descriptors of the different modules may be manually updated if necessary, please refer to [\[TARIC3-AMN\]](#) for more information on this.

The deployment of the application is done with the command:

```
$ ./configure server deploy
```

This command will deploy/assign the application in the WebLogic Server `<APP_SERVER>`³.

Another way is to do this from the WebLogic console interface.

The archive, `s_taric3` contains the business logic implementation and the presentation logic implementation for the TARIC3 system. It contains a number of J2EE modules which have to be deployed in the correct servers. The following table contains an enumeration of the J2EE modules present in the `s_taric3` archive. For each of these modules it is indicated to which server they are assigned (targets).

J2EE module	Admin server	TARIC3 server
<code>g_demco_ejb</code> This module implements a set of generic logic used by the TARIC3 application. This includes the asynchronous message handling and the business logic controller among others.	no	yes
<code>g_demco_management</code> This module is implementing the security management for the TARIC3 application.	no	yes
<code>g_demco_batch_frontend_ejb</code> This module is implementing the presentation services for the batch and reporting tasks of the TARIC3 application.	no	yes
<code>s_taric3_ejb</code> This module is implementing the business for the TARIC3 application.	no	yes
<code>s_taric_ui</code> This module is implementing the user interface for the TARIC3 application.	no	yes

³ See 4.5.1

TAXUD/R5 – TARIC3 Installation Procedure Manual	
INSTALLATION AND CONFIGURATION	Ref: TARIC3-IPR
CONFIGURATION OF THE APPLICATION	

J2EE module	Admin server	TARIC3 server
s_taric3_ug_war This module is implementing the user guide for the TARIC3 application.	no	yes
s_taric3_ob_ug_war This module is implementing the user guide for the TARIC3 Output Bridge application.	no	yes
s_taric3_ib_ug_war This module is implementing the user guide for the TARIC3 Input Bridge application.	no	yes
s_taric3_mu_ug_war This module is implementing the user guide for the TARIC3 Mass Updates application.	no	yes
s_taric3_pub_ug_war This module is implementing the user guide for the TARIC3 Publication application.	no	yes
s_taric3_rep_ug_war This module is implementing the user guide for the TARIC3 Reporting application.	no	yes

Table 2. TARIC3 J2EE Modules

4.10 Configuration of the application

Once the TARIC3 application has been successfully deployed a number of configuration tasks must be executed. These tasks are described in [\[TARIC3-AMN\]](#).

The TARIC3 WLS must run to execute these tasks. Perform the following tasks to complete the installation of the TARIC3 system:

- Load the translations as explained in section “Translation management” of [\[TARIC3-AMN\]](#);
- Configure the scheduled tasks as explained in section “Configuration of the scheduled tasks” of [\[TARIC3-AMN\]](#).

Note that these tasks must not be performed in a specific order.

4.11 Conversion Tools

This section handles the conversion of TARIC2 data to TARIC3.

4.11.1 Prerequisite

A TARIC3 schema must have been created (see 4.7). This schema is further referred to as the ‘TARIC3 destination schema’.

A dedicated schema is used during the migration from TARIC2 (in Oracle9) to TARIC3 (in Oracle10). **This schema must be created in an Oracle 10 database** and will later be referred as the ‘TARIC3 intermediate schema’.

The TARIC3 intermediate schema can be discarded after a successful migration.

TAXUD/R5 – TARIC3 Installation Procedure Manual	
INSTALLATION AND CONFIGURATION	Ref: TARIC3-IPR
CONVERSION TOOLS	

4.11.2 Customization

Edit the migration.properties file and change the following parameters:

- 'ORACLE_9_HOME': the path where to find the oracle **9.2** utilities: imp, exp and sqlplus;
- 'ORACLE_10_HOME': the path where to find the oracle **10.2** utilities: imp, exp and sqlplus.

TARIC2 Original schema parameters

- 'TARIC2_ORI': name of the TARIC2 original schema
- 'TARIC2_ORI_PWD': password to enter the TARIC2 original schema
- 'TWO_TASK_ORI': the Oracle Database SID of the TARIC2 original schema
- 'TARIC2_ORI_TABLESPACE': the name of the TARIC2 original Tablespace
- 'TARIC2_ORI_DUMP_FILE': the name of the TARIC2 dump file to create

TARIC3 Intermediate schema parameters

- 'TARIC3_INT': name of the TARIC3 intermediate schema
- 'TARIC3_INT_PWD': password to enter the TARIC3 intermediate schema
- 'TWO_TASK_INT': the Oracle Database SID of the TARIC3 intermediate schema
- 'TARIC3_INT_TABLESPACE': the name of the TARIC3 intermediate Tablespace
- 'TARIC3_INT_DUMP_FILE': the name of the TARIC3 intermediate file to create

TARIC3 Destination schema parameters

- 'TARIC3_DEST': name of the TARIC3 intermediate schema
- 'TARIC3_DEST_PWD': password to enter the TARIC3 intermediate schema
- 'TWO_TASK_DEST': the Oracle Database SID of the TARIC3 intermediate schema
- 'TARIC3_DEST_TABLESPACE': the name of the TARIC3 intermediate Tablespace

4.11.3 Conversion

The conversion of TARIC2 to TARIC3 is triggered by executing the migration.sh script.

This script executes the following sub-scripts:

- '01_exp_t2_to_intermediate.sh': exports the TARIC2 data from TARIC2 original schema and imports it in the TARIC3 intermediate schema.

TAXUD/R5 – TARIC3 Installation Procedure Manual	
INSTALLATION AND CONFIGURATION	Ref: TARIC3-IPR
SETUP OF TARIC2 COMPATIBLE REFERENCE DATABASE	

- ‘02_migration.sh’: applies some patches to the original data imported in the TARIC3 intermediate schema and starts the conversion. The data conversion is required to make *original TARIC2* data compatible with the TARIC3 schema.

The patches can be found in ‘taric2patch.sql; the migration is in ‘migration.sql’.

If the variable ‘DEBUG’ has been set to ‘ON’, then the ‘taric2patch_test.sql’ is executed. All test in this file should result to ‘number of inconsistencies: 0’, in order to approve the applied patches. For performance reasons this option can be turned of by defining ‘DEBUG’ different from ‘ON’.

- ‘03_exp_intermediate_to_t3.sh’: this is the final step in which the data is exported from the intermediate schema towards the TARIC3 destination schema. This script will also synchronize the sequences between the TARIC2 original schema and TARIC3 destination schema.

Every step can also be executed separately. This way one can intervene when something goes wrong in one of the steps before proceeding to the next step.

During the migration, a log file is produced with following name ‘migration_YYYYMMDDHHMISS.log’, where YYYY=year, MM=month, DD=Day, HH=Hour, MI=Minutes and SS=Seconds of the timestamp when the migration was launched.

4.12 Setup of TARIC2 compatible reference database

This section handles the setup of a TARIC2 compatible reference database. This database will be used in a transition phase during which external systems connecting to TARIC are not yet TARIC3 compatible.

The TARIC2 compatible reference database will contain only synonyms for TARIC2 tables used in the external systems, pointing to views deployed in the TARIC3 database. These views are simulating the TARIC2 tables by using TARIC3 tables and have the *T2_* prefix.

Before installing the synonyms, be sure that the right environmental variables have been exported. The environmental variables required are:

- **ORACLE_HOME**: this variable must be set to the installation directory of the Oracle database client.
- **PATH**: this variable must contain the \$ORACLE_HOME/bin variable.
- The value for **TWO_TASK** must be set to the name defined in tnsname for connecting to the correct Oracle instance.

To install the synonyms, create an empty database schema and execute the *create_taric2_synonyms.sql* script in the TARIC3 installation *TARIC2_views* directory.

For example (in .../install/etc/database/taric3/TARIC2_views):

```
$ sqlplus <TARIC2_user>/<TARIC2_pwd> @create_taric2_synonyms.sq
```

TAXUD/R5 – TARIC3 Installation Procedure Manual	
INSTALLATION AND CONFIGURATION	Ref: TARIC3-IPR
SETUP OF ENVIRONMENT.NAME IN THE TARIC3 GUI JNLP FILE	

The newly created TARIC2 compatible reference database must be used as a replacement for the TARIC2 database in the JDBC datasource of the external systems (see installation and configuration manual of the concerned external systems).

4.13 **Setup of `environment.name` in the TARIC3 GUI JNLP file**

In the JNLP file of the TARIC3 Swing GUI application, located in `.../s.taric3.ui/jws/launch.jnlp`, configure:

```
<property name=" environment.name" value="DEVELOPMENT"/>
```

with the correct environment name.

TAXUD/R5 – TARIC3 Installation Procedure Manual	
DETAILS OF THE WEBLOGIC RESOURCES	Ref: TARIC3-IPR
MACHINE	

5 DETAILS OF THE WEBLOGIC RESOURCES

The configuration of the TARIC3 application is done with the command:

```
$ ./configure server create (see 4.6).
```

The following sections describe in detail the resources that are created during that automatic configuration.

5.1 Machine

The name of the Machine hosting the WLS is given by the APP_HOST variable of the runtime.properties file.

5.2 Server

The name of the WLS that hosts the TARIC3 application is given by the APP_SERVER variable of the runtime.properties file.

The following remote start options for the server are set:

- The BEA home directory given by the BEA_HOME variable of the group.properties file;
- The Root directory given by the DOM_HOME variable of the group.properties file;
- The Java home directory given by the JAVA_HOME variable of the group.properties file;
- The log files locations are set.

5.2.1 Work managers

TARIC3 is currently using the default Work Manager of Weblogic 10.3 so no action is required.

5.3 Persistent stores

The following JMS JDBC Store is created:

- TARIC3Store, the database table TARIC3_WLSTORE is automatically created at the first usage of that store.

5.4 JMS servers

The following JMS Server is created and assigned to the TARIC3 WLS:

- TARIC3JMSServer with the TARIC3Store as persistent store.

The TARIC3 server is assigned as target for the JMS server.

TAXUD/R5 – TARIC3 Installation Procedure Manual	
DETAILS OF THE WEBLOGIC RESOURCES	Ref: TARIC3-IPR
JMS MODULES	

5.5 JMS Modules

5.5.1 TARIC3 JMS module

Name: TARIC3JMSModule.

The TARIC3 server is assigned as target for this JMS module.

5.5.1.1 Subdeployment

Name: TARIC3JMSSubdeployment.

The TARIC3 JMS server is assigned as target for this subdeployment.

5.5.1.2 Connection Factories

The following JMS Connection Factory is created and assigned to the TARIC3 WLS:

- TARIC3ConnectionFactory with JNDI name s/taric3/jms/ConnectionFactory;
- TARIC3BatchConnectionFactory with JNDI name s/taric3/jms/batch/ConnectionFactory;
- TARIC3ReportConnectionFactory with JNDI name s/taric3/jms/report/ConnectionFactory.

The transactional behaviour for the connection factories is configured as follow:

- XAConnection Factory Enabled: Enabled.

5.5.1.3 Required JMS queues

A set of queues is defined on the TARIC3 JMS module.

JNDI name	Name	Description
s/taric3/jms/DeadLetterQueue	TARIC3DeadLetterQueue	All messages which fail to be processed in the message beans will be transferred to this queue after Redelivery Limit is exceeded. A log entry is written each time the transfer of a message fails.
s/taric3/jms/ErrorQueue	TARIC3ErrorQueue	All messages delivered to the input queue that contain a fatal error will be put on this queue for future investigation by a system administrator. At the same time an output message for the message originator is generated and put on the corresponding output queue.
s/taric3/jms/InputQueue	TARIC3InputQueue	This is the JMS input queue for all messages asynchronously delivered to the system.
s/taric3/jms/ResultQueue	TARIC3ResultQueue	This is the JMS output queue for all JMS messages.

TAXUD/R5 – TARIC3 Installation Procedure Manual	
DETAILS OF THE WEBLOGIC RESOURCES	Ref: TARIC3-IPR
JMS MODULES	

JNDI name	Name	Description
<i>s/taric3/jms/CcnOutQueue</i>	<i>TARIC3CcnOutQueue</i>	The result of the processing of input messages originating from a CCN queue are put on this queue. All messages put onto this output queue have to be forwarded to the corresponding CCN queue by the CSI Bridge.
<i>s/taric3/jms/report/DeadLetterQueue</i>	<i>TARIC3ReportDeadLetterQueue</i>	All messages that fail to be processed in the message beans of the report module will be transferred to this queue after Redelivery Limit is exceeded. A log entry is written each time the transfer of a message fails.
<i>s/taric3/jms/report/ErrorQueue</i>	<i>TARIC3ReportErrorQueue</i>	All messages delivered to the input queue that contain a fatal error for the report part will be put on this queue for future investigation by a system administrator. At the same time an output message for the message originator is generated and put on the corresponding output queue.
<i>s/taric3/jms/report/InputQueue</i>	<i>TARIC3ReportInputQueue</i>	This is the JMS input queue for all messages asynchronously delivered to the system for the report part.
<i>s/taric3/jms/report/ResultQueue</i>	<i>TARIC3ReportResultQueue</i>	This is the JMS output queue for all JMS messages for the report part.
<i>s/taric3/jms/batch/DeadLetterQueue</i>	<i>TARIC3BatchDeadLetterQueue</i>	All messages which fail to be processed in the message beans of the batch module will be transferred to this queue after Redelivery Limit is exceeded. A log entry is written each time the transfer of a message fails.
<i>s/taric3/jms/batch/ErrorQueue</i>	<i>TARIC3BatchErrorQueue</i>	All messages delivered to the input queue that contain a fatal error for the batch part will be put on this queue for future investigation by a system administrator. At the same time an output message for the message originator is generated and put on the corresponding output queue.
<i>s/taric3/jms/batch/InputQueue</i>	<i>TARIC3BatchInputQueue</i>	This is the JMS input queue for all messages asynchronously delivered to the system for the batch part.
<i>s/taric3/jms/batch/ResultQueue</i>	<i>TARIC3BatchResultQueue</i>	This is the JMS output queue for all JMS messages for the batch part.
<i>s/taric3/jms/batch/TimerQueue</i>	<i>TARIC3BatchTimerQueue</i>	This is the JMS timer queue used to simulate a timer service.
<i>s/taric3/jms/batch/ReadyToRunLowQueue</i>	<i>TARIC3BatchReadyToRunLowQueue</i>	This is the JMS queue on which jobs with a low priority that are ready to run are posted.
<i>s/taric3/jms/batch/ReadyToRunMediumQueue</i>	<i>TARIC3BatchReadyToRunMediumQueue</i>	This is the JMS queue on which jobs with a medium priority that are ready to run are posted.

TAXUD/R5 – TARIC3 Installation Procedure Manual	
DETAILS OF THE WEBLOGIC RESOURCES	Ref: TARIC3-IPR
JMS MODULES	

JNDI name	Name	Description
<i>s/taric3/jms/batch/ReadyToRunHighQueue</i>	<i>TARIC3BatchReadyToRunHighQueue</i>	This is the JMS queue on which jobs with a high priority that are ready to run are posted.
<i>s/taric3/jms/inb/DeadLetterQueue</i>	<i>TARIC3InputBridgeDeadLetterQueue</i>	All messages which fail to be processed in the TARIC3 Input Bridge message beans will be transferred to this queue after Redelivery Limit is exceeded. A log entry is written each time the transfer of a message fails.
<i>s/taric3/jms/inb/ErrorQueue</i>	<i>TARIC3InputBridgeErrorQueue</i>	All messages delivered to the TARIC3 Input Bridge RequestQueue that contain a fatal error will be put on this queue for future investigation by a system administrator. At the same time an output message for the message originator is generated and put on the corresponding output queue.
<i>s/taric3/jms/inb/RequestQueue</i>	<i>TARIC3InputBridgeRequestQueue</i>	This is the JMS input queue for all messages asynchronously delivered to the TARIC3 Input Bridge system.
<i>s/taric3/jms/inb/ResponseQueue</i>	<i>TARIC3InputBridgeResponseQueue</i>	This is the JMS output queue for all JMS messages handled by the TARIC3 Input Bridge system.

Table 3. Required JMS queues for the TARIC3JMSServer

5.5.1.4 JMS queues properties

For all queues, the subdeployment property is set to:

TARIC3JMSSubdeployment (defined in section 5.5.1.1).

For each of the queues, except all the *DeadLetterQueue* queues and the *TimerQueue*, the following properties are configured:

Property name	Property value
<i>Redelivery Limit</i>	The value “3” is set for this property. As the handling of the message does not contain any data dependent logic, there is no real reason to believe that a message delivery will fail.
<i>Error Destination</i>	If the <i>Redelivery Limit</i> is reached, the message will be transferred to this queue in order to not block the correct handling of other messages. The property is set to the corresponding dead letter queue.

Table 4. TARIC3 JMS queues properties

For the *TimerQueue* the following properties are configured:

Property name	Property value
<i>Redelivery Delay Override</i>	The value “60000” is set for this property, allowing some time to fix the error which caused the redelivery.
<i>Redelivery Limit</i>	The value “1000000” is set for this property. As the handling of the message does not contain any data dependent logic, there is no real

TAXUD/R5 – TARIC3 Installation Procedure Manual	
DETAILS OF THE WEBLOGIC RESOURCES	Ref: TARIC3-IPR
JMS MODULES	

Property name	Property value
	reason to believe that a message delivery will fail. But if it fails, redelivery has to be tried. The value is set this high to prevent the final failure of the message delivery, which will result in the failure of the timer service based on this queue. As a consequence, the application should be redeployed to restart the timer service.
<i>Error Destination</i>	If the <i>Redelivery Limit</i> is reached, the message will be transferred to this queue in order to not block the correct handling of other messages. This property is set to the corresponding dead letter queue.

Table 5. TARIC3 JMS queues properties for Timer Queue

5.5.1.5 Required JMS topics

A set of topics is created on the TARIC3 JMS module.

JNDI name	Name	Description
<i>s/taric3/jms/batch/TimerTopic</i>	TARIC3BatchTimerTopic	This is the JMS timer topic which periodically receives a timer message (a 'tick') after a configurable period (e.g. every minute ...).
<i>s/taric3/jms/batch/JobNotificationTopic</i>	TARIC3BatchJobNotificationTopic	This is the JMS job notification topic, which will receive a message for each finished job.

Table 6. Required JMS topics for the TARIC3JMSServer

5.5.1.6 JMS topic properties

For the *JobNotificationTopic*, the following properties are configured:

Property name	Property value
<i>Subdeployment</i>	<i>TARIC3JMSSubdeployment</i> defined in section 5.5.1.1
<i>Redelivery Limit</i>	The value "3" is set for this property. As the handling of the message does not contain any data dependent logic, there is no real reason to believe that a message delivery will fail.
<i>Error Destination</i>	If the <i>Redelivery Limit</i> is reached, the message will be transferred to this queue in order to not block the correct handling of other messages. This property is set to the corresponding dead letter queue.

Table 7. TARIC3 JMS topic properties

For the *TimerTopic*, the following properties are configured:

Property name	Property value
<i>Subdeployment</i>	<i>TARIC3JMSSubdeployment</i> defined in section 5.5.1.1
<i>Redelivery Limit</i>	It is advised to put a value of 0 for this property since no redelivery should be allowed.
<i>Error Destination</i>	If the <i>Redelivery Limit</i> is reached, the message will be transferred to this queue in order to not block the correct handling of other messages. This property is set to the corresponding dead letter queue.

Table 8. TARIC3 JMS topic properties for Timer Topic

TAXUD/R5 – TARIC3 Installation Procedure Manual	
DETAILS OF THE WEBLOGIC RESOURCES	Ref: TARIC3-IPR
JDBC DATA SOURCES	

5.6 JDBC data sources

Four data sources are created that have access to the TARIC3 tables.

5.6.1 XA data sources

One XA Data Source is created. The following properties are assigned for this data source:

Property name	Property value
<i>Name</i>	<i>TARIC3DataSource</i>
<i>JNDI Name</i>	<i>s/taric3/jdbc/TxDataSource</i>
<i>Database Type</i>	Oracle
<i>Database Driver</i>	Oracle Driver (Thin XA) Versions 9.0.1, 9.2.0, 10, 11
<i>Database name</i>	<i><instance_name></i> This value is taken from the variable DB_CONN_URL of the runtime.properties file.
<i>Host name</i>	<i><host></i> This value is taken from the variable DB_CONN_URL of the runtime.properties file.
<i>Port</i>	<i><port></i> This value is taken from the variable DB_CONN_URL of the runtime.properties file.
<i>Database user name</i>	<i><oracle-uid></i> This user is taken from the variable USER of the runtime.properties file.
<i>Password</i>	<i><password></i> This user is taken from the variable PASS of the runtime.properties file.
<i>Configuration/General/Row Prefetch Enabled</i>	True
<i>Configuration/General/Row Prefetch Size</i>	48
<i>Configuration/General/Stream Chunk Size</i>	16384
<i>Configuration/Connection Pool/Initial Capacity</i>	0
<i>Configuration/Connection Pool/Maximum Capacity</i>	20
<i>Configuration/Connection Pool/Capacity Increment</i>	1
<i>Configuration/Connection Pool/Test Connections on Reserve</i>	True
<i>Configuration/Connection Pool/Test Table Name</i>	SQL SELECT 1 FROM DUAL
<i>Configuration/Connection Pool/Shrink Period</i>	900 (In seconds)
<i>Configuration/Transaction/Set XA Transaction Timeout</i>	True

TAXUD/R5 – TARIC3 Installation Procedure Manual	
DETAILS OF THE WEBLOGIC RESOURCES	Ref: TARIC3-IPR
JDBC DATA SOURCES	

Property name	Property value
<i>Configuration/Transaction/XA Transaction Timeout</i>	1500

Table 9. TARIC3 XA data source

The TARIC3 server is assigned as target for this Data Source.

5.6.2 Non-XA Data sources

One Data Source is created. The following properties are assigned for this data source:

Property name	Property value
<i>Name</i>	<i>TARIC3NonXADataSource</i>
<i>JNDI Name</i>	<i>s/taric3/jdbc/batch/TxDataSource</i>
<i>Database Type</i>	Oracle
<i>Database Driver</i>	Oracle Driver (Thin) Versions 9.0.1, 9.2.0, 10, 11
<i>Support global transactions</i>	True
<i>Emulate Two-Phase commit</i>	True
<i>Database name</i>	<i><instance_name></i> This value is taken from the variable DB_CONN_URL of the runtime.properties file.
<i>Host name</i>	<i><host></i> This value is taken from the variable DB_CONN_URL of the runtime.properties file.
<i>Port</i>	<i><port></i> This value is taken from the variable DB_CONN_URL of the runtime.properties file.
<i>Database user name</i>	<i><oracle-uid></i> This user is taken from the variable USER of the runtime.properties file.
<i>Password</i>	<i><password></i> This user is taken from the variable PASS of the runtime.properties file.
<i>Configuration/General/Row Prefetch Enabled</i>	True
<i>Configuration/General/Row Prefetch Size</i>	48
<i>Configuration/General/Stream Chunk Size</i>	16384
<i>Configuration/Connection Pool/Initial Capacity</i>	0
<i>Configuration/Connection Pool/Maximum Capacity</i>	20
<i>Configuration/Connection Pool/Capacity Increment</i>	1
<i>Configuration/Connection Pool/Test Connections on Reserve</i>	True
<i>Configuration/Connection Pool/Test Table Name</i>	SQL SELECT 1 FROM DUAL
<i>Configuration/Connection</i>	900 (In seconds)

TAXUD/R5 – TARIC3 Installation Procedure Manual	
DETAILS OF THE WEBLOGIC RESOURCES	Ref: TARIC3-IPR
TARIC3 SERVER JNDI TREE VALIDATION	

Property name	Property value
<i>Pool/Shrink Period</i>	

Table 10. TARIC3 non-XA data source

The TARIC3 server is assigned as target for this Data Source.

5.7 TARIC3 server JNDI tree validation

Verify the JNDI tree for the TARIC3 server by selecting the server in the WebLogic console (right click/view JNDI tree). The following entries should be present:

JNDI Entry	Purpose
<i>s/taric3/ejb/ACLManager</i>	The Access Control List manager bean.
<i>s/taric3/ejb/admin/Authenticator</i>	The EJB implementing the user management services.
<i>s/taric3/ejb/admin/CcnUserManagement</i>	The EJB implementing the user impersonation.
<i>s/taric3/ejb/admin/WLSSupport</i>	The EJB implementing some support functionality <i>for the CcnUserManagement</i> bean.
<i>s/taric3/ejb/batch/BusinessController</i>	The business controller EJB of the batch module.
<i>s/taric3/ejb/batch/BusinessControllerMultipart</i>	Idem but multi-part.
<i>s/taric3/ejb/batch/EnvelopeHandler</i>	The asynchronous message envelope handler of the batch module.
<i>s/taric3/ejb/batch/JobAgent</i>	The job agent bean.
<i>s/taric3/ejb/batch/JobManager</i>	The job manager bean.
<i>s/taric3/ejb/batch/JobManagerMultipart</i>	Idem but multi-part.
<i>s/taric3/ejb/batch/PrivateUpdate</i>	A support bean for the batch module.
<i>s/taric3/ejb/batch/TransactionHandler</i>	A support bean used in the asynchronous message handler to manage the transaction elements, for the batch module.
<i>s/taric3/ejb/batch/TransactionHandlerMultipart</i>	Idem but multi-part.
<i>s/taric3/ejb/batch/ValidationHelper</i>	A support bean for the validation of job parameters.
<i>s/taric3/ejb/batch/WithoutTransaction</i>	A support bean for the batch module, responsible for starting jobs inside their own transaction.
<i>s/taric3/ejb/batch/WithTransaction</i>	A support bean for the batch module, responsible for starting jobs outside the main transaction.
<i>s/taric3/ejb/BusinessController</i>	The business controller EJB.
<i>s/taric3/ejb/BusinessTransaction</i>	The EJB responsible for creating a new business transaction.
<i>s/taric3/ejb/Configuration</i>	The configuration bean.
<i>s/taric3/ejb/DictionaryManager</i>	The translation management EJB.
<i>s/taric3/ejb/EnvelopeHandler</i>	The asynchronous message envelope handler.
<i>s/taric3/ejb/FrontEndSupport</i>	The presentation tier support EJB.
<i>s/taric3/ejb/IOLocal</i>	Utility bean to exchange files.
<i>s/taric3/ejb/IORemote</i>	Utility bean to exchange files.
<i>s/taric3/ejb/QueryPersistence</i>	The query persistence bean.
<i>s/taric3/ejb/report/BusinessController</i>	The business controller EJB for the reporting service.

TAXUD/R5 – TARIC3 Installation Procedure Manual	
DETAILS OF THE WEBLOGIC RESOURCES	Ref: TARIC3-IPR
TARIC3 SERVER JNDI TREE VALIDATION	

JNDI Entry	Purpose
<i>s/taric3/ejb/report/EnvelopeHandler</i>	The asynchronous message envelope handler of the reporting module.
<i>s/taric3/ejb/report/Reporting</i>	The reporting module EJB.
<i>s/taric3/ejb/report/TransactionHandler</i>	A support bean used in the asynchronous message handler to manage the transaction elements, for the reporting module.
<i>s/taric3/ejb/ResultStorage</i>	A support bean used in the asynchronous message handler to temporary store transaction results.
<i>s/taric3/ejb/TransactionHandler</i>	A support bean used in the asynchronous message handler to manage the transaction elements.
<i>s/taric3/ejb/UserProfile</i>	The country membership controller EJB.
<i>s/taric3/ejb/UUID</i>	The EJB that generates UUID's.
<i>s/taric3/jdbc/TxDatasource</i>	The XA enabled data source for the TARIC3 tables.
<i>s/taric3/jdbc/batch/TxDataSource</i>	The non-XA enabled data source for the TARIC3 tables.
<i>s/taric3/jdbc/report/TxDataSource</i>	The non-XA enabled data source used by the reporting system for the TARIC3 tables.
<i>s/taric3/jms/batch/DeadLetterQueue</i>	All messages which fail to be processed in the message beans of the batch module will be transferred to this queue after Redelivery Limit is exceeded. A log entry is written each time the transfer of a message fails.
<i>s/taric3/jms/batch/ConnectionFactory</i>	The JMS connection factory used by the batch system to access the queues hosted on the different JMS Server.
<i>s/taric3/jms/batch/ErrorQueue</i>	All messages delivered to the input queue that contain a fatal error for the batch part will be put on this queue for future investigation by a system administrator. At the same time an output message for the message originator is generated and put on the corresponding output queue.
<i>s/taric3/jms/batch/InputQueue</i>	This is the JMS input queue for all messages asynchronously delivered to the system for the batch part.
<i>s/taric3/jms/batch/JobNotificationTopic</i>	This is the JMS job notification topic, which will receive a message for each finished job.
<i>s/taric3/jms/batch/ReadyToRunHighQueue</i>	This is the JMS queue on which jobs with a high priority that are ready to run are posted.
<i>s/taric3/jms/batch/ReadyToRunLowQueue</i>	This is the JMS queue on which jobs with a low priority that are ready to run are posted.
<i>s/taric3/jms/batch/ReadyToRunMediumQueue</i>	This is the JMS queue on which jobs with a medium priority that are ready to run are posted.
<i>s/taric3/jms/batch/ResultQueue</i>	This is the JMS output queue for all JMS messages for the batch part.
<i>s/taric3/jms/batch/TimerQueue</i>	This is the JMS timer queue used to simulate a timer service.
<i>s/taric3/jms/batch/TimerTopic</i>	This is the JMS timer topic which periodically

TAXUD/R5 – TARIC3 Installation Procedure Manual	
DETAILS OF THE WEBLOGIC RESOURCES	Ref: TARIC3-IPR
TARIC3 SERVER JNDI TREE VALIDATION	

JNDI Entry	Purpose
	receives a timer message (a 'tick') after a configurable period (e.g. every minute ...).
<i>s/taric3/jms/CcnOutQueue</i>	The result of the processing of input messages originating from a CCN queue are put on this queue. All messages put onto this output queue have to be forwarded to the corresponding CCN queue by the CSI Bridge.
<i>s/taric3/jms/ConnectionFactory</i>	The JMS connection factory to access the queues hosted on the different JMS Server.
<i>s/taric3/jms/DeadLetterQueue</i>	All messages which fail to be processed in the message beans will be transferred to this queue after Redelivery Limit is exceeded. A log entry is written each time the transfer of a message fails.
<i>s/taric3/jms/ErrorQueue</i>	All messages delivered to the input queue that contain a fatal error will be put on this queue for future investigation by a system administrator. At the same time an output message for the message originator is generated and put on the corresponding output queue.
<i>s/taric3/jms/InputQueue</i>	This is the JMS input queue for all messages asynchronously delivered to the system.
<i>s/taric3/jms/report/ConnectionFactory</i>	The JMS connection factory used by the reporting system to access the queues hosted on the different JMS Server.
<i>s/taric3/jms/report/DeadLetterQueue</i>	All messages that fail to be processed in the message beans of the report module will be transferred to this queue after Redelivery Limit is exceeded. A log entry is written each time the transfer of a message fails.
<i>s/taric3/jms/report/ErrorQueue</i>	All messages delivered to the input queue that contain a fatal error for the report part will be put on this queue for future investigation by a system administrator. At the same time an output message for the message originator is generated and put on the corresponding output queue.
<i>s/taric3/jms/report/InputQueue</i>	This is the JMS input queue for all messages asynchronously delivered to the system for the report part.
<i>s/taric3/jms/report/ResultQueue</i>	This is the JMS output queue for all JMS messages for the report part.
<i>s/taric3/jms/ResultQueue</i>	This is the JMS output queue for all JMS messages.
<i>s/taric3/web/ejb/batch/frontend/FrontEnd</i>	Web front-end bean for the batch system.
<i>s/taric3/web/ejb/batch/frontend/FrontEndSupport</i>	Support bean for the batch system.
<i>s/taric3/web/ejb/batch/frontend/MultipartSync2Async</i>	Support bean for the batch system.

Table 11. TARIC3 JNDI tree

TAXUD/R5 – TARIC3 Installation Procedure Manual	
DIRECTORY CONTENT	Ref: TARIC3-IPR
TARIC3 SERVER JNDI TREE VALIDATION	

6 DIRECTORY CONTENT

The following table describes the content of the directories in the <GROUP> hierarchy. These are used for configuration and at runtime of both the server(s) and the application(s) that are part of this group.

Name	Description
<i>group.properties</i>	Common configuration file for the servers and applications installed in this group
<i>app</i>	Directory containing one subdirectory for each application installed in the current group. It contains a series of application specific subdirectories and files
<i>bin</i>	Binaries and shell script used by the applications
<i>log</i>	Output directory for WLS logs
<i>tlog</i>	Output directory for WLS transaction logs, these files should not be modified or deleted
<i>jmspaging</i>	Directory where the JMS paging information will be saved.

Table 12. Content of an application group hierarchy

The following table describes the content of the directories used by an application installed in a group. These are used for configuration of the application and the server that hosts it, and at runtime by the application components and its server.

Name	Description
<i>ant</i>	Contains a distribution of the ant utility. Required for configuration scripts.
<i>runtime.properties</i>	Runtime configuration file for the application
<i>configure</i>	Configuration script for application and its server
<i>bin</i>	Binaries and shell script used by the application
<i>etc</i>	Additional files for the application, including templates for scripts
<i>data</i>	Data directory for the application
<i>database</i>	Database configuration scripts for the application
<i>j2ee_archives</i>	J2EE archives to deploy the application
<i>exploded</i>	Contains an exploded distribution of the J2EE application archive.
<i>util_archives</i>	utility archives for the application
<i>tmp</i>	Temporary workspace
<i>log</i>	Output directory for log files

Table 13. Content of an application directory hierarchy