Banking Application Installation Guide

Version 2005, Rev. B

May 2010

Contents

5   WebSphere 6.1 Installation Process

6   Installing Third Party Software

7   Starting the Banking Applications

8   Installing the Branch Offline Server

9 Siebel Retail Finance Integration with Siebel 7.x

1 What's New in This Release

What's New in Banking Application Installation Guide, Version 2005, Rev. B

Topic	Description
Chapter 5 WebSphere 6.1 Installation Process	Information on WebSphere 6.1 installation is presented here.

2 The Banking Application

The Siebel Retail Finance Banking Application consists of the following components:

- A database containing the data required by the product.

- A deployable Enterprise Application Archive (EAR) file, which contains:

    - The standard server-side functionality of the Banking Application, which is deployed on a Central Server.

    - Java Swing-based client and administration applications for Branch Teller.

    - Web applications for Internet Banking and MCA Services administration functionality.

- A Branch Offline Server

- Configuration files for integration with Siebel Financial Services 7.x

This document describes how to:

- Create and configure the database

- Deploy the Banking Application EAR file

- Configure printing

- Deploy and configure the Branch Offline Server

- Start the Banking applications

- Integrate the Banking Application with Siebel Financial Services 7.x

3   WebLogic 8.1 Installation Process

This chapter describes how to install the Banking Application on the WebLogic platform. It contains the following topics:

- Requirements for Installing the Banking Application on Windows
- Requirements for Installing the Banking Application on UNIX
- Extracting the SRF Banking Application Files
- Setting Up the Oracle Database
- Deploying the Banking Application on WebLogic

NOTE: Refer to the Siebel Retail Finance System Requirements and Supported Platforms document on the Siebel SupportWeb Web site (http://supportweb.siebel.com) for information regarding the supported environments, including the supported database and application server versions, and supported versions of third- party software. Your Siebel TAM will have provided you with a SupportWeb logon.

Requirements for Installing the Banking Application on Windows

The installation prerequisites for installing on Windows are as follows:

- The target machine must be clean, that is, not running any other WebLogic applications, including any previous version of the Banking Application.
- BEA WebLogic 8.1.0 must be installed and configured. This guide assumes that the WebLogic root directory is: D:\bea\weblogic81.
- Oracle must be installed and configured. This guide assumes that Oracle is installed at: D:\oracle.
- The Java utilities java, javac, and jar must be available at the command line.
- This guide assumes the CD-ROM drive is attached to drive: E:\.
- Siebel Financial Services 7.x must be installed, if it is required to interface with that product.

Requirements for Installing the Banking Application on UNIX

The installation prerequisites for installing on UNIX are as follows:

- The target machine must be clean, that is, not running any other WebLogic applications, including any previous version of the Banking Application.

■ BEA WebLogic 8.1.0 must be installed and configured. This guide assumes that the WebLogic root directory is: /bea/weblogic81.

■ Oracle must be installed and configured. $ORACLE_HOME must be configured to the Oracle installation location.

■ The Java utilities java, javac, and jar must be available at the command line.

■ This guide assumes the CD-ROM drive is mounted at: /mnt/cdrom.

■ Siebel Financial Services 7.x must be installed, if it is required to interface with that product.

Extracting the SRF Banking Application Files

The Banking Application files are located on the Banking Application CD for WebLogic/Oracle in the **packs\SRFBankingApplication***vx.x***WebLogicOracle.jar** file. You must extract this file into the local file system (drive D:\ is assumed throughout this document – replace as appropriate).

NOTE: All instances of *vx.x* in the instructions in this document refer to the version of the software that you are using.

To extract the files on Windows

1 Open a command prompt.

2 Enter the following commands:

    cd /d d:\

    jar xvf e:\packs\SRFBankingApplication*vx.x*WebLogicOracle.jar

This command creates the **siebel** folder in **d:\.**

To extract the files on UNIX

1 Open a command shell.

2 Enter the following commands:

    cd /tmp

    jar xvf /mnt/cdrom/packs/SRFBankingApplication*vx.x*WebLogicOracle.jar

This creates the **siebel** folder in **/tmp.**

Setting Up the Oracle Database

Create a database on your Oracle database server machine, using all the default settings, and create a user with DBA rights on this database. For the purpose of these instructions it is assumed the database is called bankfrm and the user is also called bankfrm. See your Oracle documentation for information on how to create databases and users.

Oracle Database Prerequisites

■   The Oracle command line utilities must be available at the command line.

■   You must create an Oracle database and make a note of the name you give it.

■   You must create an Oracle user for the database. Make a note of the user ID and password.

■   You must create an entry for the database in the tnsnames.ora file for the server on which the application is to be deployed.

Importing the Oracle Dump File

The data for the Banking Application database is provided as an Oracle dump file, as some of the tables contain RAW data that you cannot import using standard SQL insert scripts.

When you import the dump file, the existing database is first dropped and then the database is loaded with the contents of the `\siebel\database\staging.dmp` file. The first time you run the ant drop or ant load commands, you are prompted for a database user name, a database password, and a database instance or db name. The process creates a `database.properties` file at `D:\temp\build_tmp\`. This `database.properties file` determines what database is used by the ant drop and ant load processes. If you need to use a different database, configure the database.properties file accordingly.

In this section the following are used:

■   The database name is bankfrm.

■   The database user ID is bankfrm.

■   The database password is bankfrm.

Adjust these values based on the database name, user ID, and password you chose in the previous section.

To import the Oracle dump file

1   Type the command `cd /d d:\siebel\deploy\database`.

2   To drop the database if it already exists type the command `ant drop`.

3   To load the database type the command `ant load`.

To reset the Oracle database

■   Type the command `ant reset-branches`.

Deploying the Banking Application on WebLogic

If you are deploying the Banking Application on a single WebLogic node (that is, not in a clustered environment), you can use the deployment script in the \siebel\deploy folder. This process is described in Deploying the Banking Application on a Single WebLogic Node on page 16.

If you are deploying the Banking Application to a clustered environment, you must follow the manual process described in Deploying the Banking Application in a Clustered WebLogic Environment on page 19.

Deploying the Banking Application on a Single WebLogic Node

Before you run the deployment script, you must configure the build.properties file, located in the `\siebel\deploy` folder, with the correct values for your WebLogic server.

To configure the build.properties file

1   Open the build. properties file in a text editor.

2   Change the server.port setting to:

`server.port = port`

where *port* is the port on which the WebLogic server listens.

3   Change the ear.path setting to:

`ear.path = path`

Where *path* is the path to the EAR file to be deployed; this is normally only necessary when deploying on Solaris.

Running the Deployment Script

The first time you run the install command, you are prompted for a database user name, a database password, and a database instance or db name. The process creates a `database.properties` file at `D:\temp\build_tmp\`. This database.properties file determines which database is used by the ant drop and ant load processes. If you need to use a different database, configure the database.properties file accordingly. You are not prompted for these details if you have already imported the Oracle database.

The deployment script automatically creates a new WebLogic server instance located at `d:\bea\user_projects\siebel`, or `/bea/user_projects/siebel`, and deploys the EAR file to this instance.

To run the deployment script on Windows

1   Open a command prompt.

2   Navigate to the d:\siebel\deploy directory.

3   Enter the following command:

```
install D:\bea
```

To run the deployment script on Solaris

1 Make sure you have updated the `ear.path` setting in build.properties to point to the location on the Solaris machine where you have copied the EAR file.

2 Open a command shell, and enter the following at the command prompt:

```
cd /tmp/siebel/deploy

chmod +x install.sh

./install.sh /bea
```

Configuring the BankframeResource.properties File

If you changed the port on which the WebLogic server is listening, you must make adjustments to the BankframeResource.properties file. If you have not changed the `server.port` setting (from its default of **7001**), you can skip this section.

To configure the BankframeResource.properties file

1 Open the following file in a text editor:

- On Windows XP:
  d:\bea\user_projects\siebel\siebel.ear\resources\BankframeResource.properties

- On Solaris: /bea/user_projects/siebel/siebel.ear/resources/BankframeResource.properties

2 Use the text editor to replace all occurrences of the string: **7001** with the value of the port on which you have configured WebLogic to listen.

3 Search for the string `ejb.initialContextFactory=` and replace all text to the right of this string with `weblogic.jndi.T3InitialContextFactory`, which is the WebLogic-specific context setting for WebLogic versions 4.51, 5.1, 6.1, and higher.

4 Remove the comment (#) for the following SSL settings for WebLogic:

```
# WebLogic

#channel.https.ssl.protocol=com.sun.net.ssl.internal.www.protocol

#channel.https.ssl.provider=com.sun.net.ssl.internal.ssl.Provider
```

5 Save the file.

Configuring the Database Drivers

You must update the startsiebel script with the path that WebLogic uses to locate the database drivers.

By default, the startsiebel script is configured for a USASCII database. You must update the script to use the UTF8 character set.

To configure the database drivers on Windows

1   Open the `\bea\user_projects\siebel\startsiebel.cmd` file in a text editor.

2   For an Oracle character set, add the following line at the start of the startsiebel.cmd file:

```
chcp 1252

set NLS_LANG=AMERICAN_AMERICA.AL32UTF8
```

3   For Oracle 9.2.0, add the following line before the Java command at the end of the file:

```
set PATH=%WL_HOME%\server\bin\oci920_8;%PATH%
```

4   Save the file.

To configure the database drivers on Solaris

1   Open the `/bea/user_projects/siebel/startsiebel.sh` file in a text editor.

2   For an Oracle character set, add the following line at the start of the startsiebel.sh file:

```
NLS_LANG=AMERICAN_AMERICA.AL32UTF8

export NLS_LANG
```

3   After the line beginning with `WL_HOME=` add the following to set your Java Home:

```
JAVA_HOME=$WL_HOME/../jdk141_02; export JAVA_HOME

PATH=$JAVA_HOME/bin:$PATH; export PATH
```

4   For Oracle 9.2.0, add the following line before the Java command at the end of the file:

```
set LD_LIBRARY_PATH=$WL_HOME/server/lib/solaris/oci920_8:$LD_LIBRARY_PATH;

export LD_LIBRARY_PATH
```

5   Save the file.

Starting the Application Server

After you deploy the Banking Application, you must start the application server.

To start the application server on Windows

1   Open a command prompt.

2   Navigate to `D:\bea\user_projects\siebel`

3   Enter the following command:

```
startsiebel
```


To start the application server on Solaris

1   Open a command shell.

2   Enter the following at the command prompt:

```
cd /bea/user_projects/siebel

chmod +x ./startsiebel.sh

nohup ./startsiebel.sh &
```

Deploying the Banking Application in a Clustered WebLogic Environment

To deploy the Banking Application on a WebLogic cluster you must use the WebLogic administration console. The following subtopics assume that your WebLogic cluster is configured as follows:

- There is one Administration Server configured, which manages the cluster, and this server is named adminSiebelServer.

- There is one proxy server configured to load balance requests across all nodes in the cluster, and this server is named siebelProxyServer.

- There are two or more managed servers, which are members of the cluster. The cluster is named siebelCluster1, and the individual nodes on the cluster are named siebelServer1 to siebelServer*n*, where *n* is the number of nodes in the cluster.

- There is a running Node Manager instance on each machine where nodes are located.

Refer to your WebLogic documentation for information about how to configure the WebLogic cluster.

Configuring the Database Drivers

You must update the startsiebel script with the path that WebLogic uses to locate the database drivers.

By default, the startsiebel script is configured for a USASCII database. You must update the script to use the UTF8 character set.

To configure the database drivers on Windows

1   Open in a text editor the *adminserverhome* /`startsiebel.cmd` file used to start the Administration Server, where *adminserverhome* is the home folder of the Administration Server.

2   For an Oracle character set, add the following line at the start of the `startsiebel.cmd` file:

```
chcp 1252

set NLS_LANG=AMERICAN_AMERICA.AL32UTF8
```

3   For Oracle 9.2.0, add the following line before the Java command at the end of the file:

```
set PATH=%WL_HOME%\server\bin\oci920_8;%PATH%
```

4   Save the file.

5   Repeat steps 1 through 4 for the *BEA_HOME*/weblogic81/server/bin startNodeManager.cmd script that you use to start the Node Manager.

To configure the database drivers on Solaris

1   Open in a text editor the *adminserverhome* `/startsiebel.sh` file used to start the Administration Server, where *adminserverhome* is the home folder of the Administration Server.

2   For an Oracle character set, add the following line at the start of the `startsiebel.sh` file:

`NLS_LANG=AMERICAN_AMERICA.AL32UTF8`

`export NLS_LANG`

3   After the line beginning with `WL_HOME=` add the following to set your Java Home:

`JAVA_HOME=$WL_HOME/../jdk141_02; export JAVA_HOME`

`PATH=$JAVA_HOME/bin:$PATH; export PATH`

4   For Oracle 9.2.0, add the following line before the Java command at the end of the file:

`set LD_LIBRARY_PATH=$WL_HOME/server/lib/solaris/oci920_8:$LD_LIBRARY_PATH;`

`export LD_LIBRARY_PATH`

5   Save the file.

6   Repeat steps 1 through 5 for the *BEA_HOME*/weblogic81/server/bin/startNodeManager.sh script that you use to start the Node Manager.

Configuring the Database Connection

To configure the database connection, you must configure a database pool and data source.

Before you configure the database connection, make sure that the Administration Server and the Node Manager are running. Refer to the WebLogic documentation for information about how to start the Node Manager and Administration Server.

To configure the database connection

1   Start the WebLogic administration console.

2   Navigate to Services > JDBC > Connection Pools in the navigation tree.

3   Select Configure a new JDBC Connection.

4   In the Database Type field, select Oracle and in the Database Driver field, select WebLogic's Oracle Driver (Type 2 XA)…, then click Continue.

5   In the Name field, type bankframePool, enter the connection details for the database that you want to connect to, and click Continue.

6   Click Test Driver Configuration to confirm that the database connection is correctly configured.

7   Select All servers in the cluster and click Create and deploy.

8   Navigate to Services > JDBC > Data Sources in the navigation tree.

9   Select Configure a new JDBC Data Source.

10   In the Name and JNDI Name fields, type bankfrm, and click Continue.

11   In the Pool Name field, select bankframePool and click Continue.

12 Select All servers in the cluster and click Create.

Deploying the Banking Application

Before you deploy the Banking Application, make sure that the Administration Server and the Node Manager are running. Refer to the WebLogic documentation for information about how to start the Node Manager and Administration Server.

Also make sure that you have uploaded the siebel.ear file to the location on the machine on which the Administration Server is running.

To deploy the Banking Application

1   Start the WebLogic Administration Console.

2   Navigate to Deployments > Applications in the navigation tree.

3   Select Deploy a new Application.

4   Navigate to the location in which you have placed the EAR file and click the radio button beside the EAR file.

5   Click Target Application.

6   Select All servers in the cluster, and click Continue.

7   Select the option Copy this application onto every target for me, and click Deploy.

Updating the BankframeResource.properties File

Before you can access the Banking Application you must update the BankframeResource.properties file for each node in the cluster so that the ejb.server setting points to the correct node.   The file is located in:

adminserverhome/*nodename*/`stage/siebel/resources/`

where *adminserverhome* is the home folder of the Administration Server (the same folder in which startsiebel.sh is located) and *nodename* is the node name, for example, siebelServer1.

Edit the BankframeResource.properties file for each node, and change the ejb.server setting to:

`ejb.server=t3://`*nodename*`:`*portnumber*

where *nodename* is the node name and *portnumber* is the port number on which the node is listening.

Starting the Cluster

Refer to the WebLogic documentation for information about how to start the Node Manager, Administration Server, and the cluster. When you have started the Node Manager, Administration Server, and cluster, the deployment of the Banking Application on the cluster is complete.

4 WebSphere 5.1.1 Installation Process

This chapter describes how to install the Banking Application on the WebSphere platform. It contains the following topics:

■ Requirements for Installing the Banking Application on Windows

■ Requirements for Installing the Banking Application on UNIX

■ Deploying the Banking Application in a WebSphere Clustered Environment

■ Extracting the SRF Banking Application Files

■ Configuring WebSphere Database Settings

■ Configuring WebSphere JMS Settings

■ Configuring Pass by Reference

■ Configuring the BankframeResource.Properties File

■ Deploying the Banking Application on WebSphere 5.1.1

NOTE: Refer to the Siebel Retail Finance System Requirements and Supported Platforms document on the Siebel SupportWeb Web site (http://supportweb.siebel.com) for information regarding the supported environments, including the supported database and application server versions, and supported versions of third- party software. Your Siebel TAM will have provided you with a SupportWeb logon.

Requirements for Installing the Banking Application on Windows

This section outlines the installation prerequisites when installing the Banking Application on Windows. This guide assumes the installation locations in the following list; adjust the values in the examples to your machine configuration.

■ The target server must be clean, that is, not running any other WebSphere applications, including any previous version of the Banking Application.

■ IBM WebSphere 5.1.1 must be installed and configured. This guide assumes that the WebSphere root directory is: `D:\Program Files\WebSphere.`

■ A database server must be installed and configured; this release supports both DB2 and Oracle. This guide assumes that:

  ■ DB2 is installed at: `D:\SQLLIB.`

  ■ Oracle is installed at: `D:\Oracle.`

■ The Java utilities `java`, `javac`, and `jar` must be available at the command line.

■ This guide assumes that the CD-ROM drive is attached to drive `E:\`; adjust the values in the examples to your machine configuration.

- If interfacing with Siebel Financial Services 7.x is required an installation of Siebel Financial Services 7.x is a prerequisite.

Requirements for Installing the Banking Application on UNIX

This section outlines the installation prerequisites when installing the Banking Application on UNIX. This guide assumes the installation locations listed below; adjust the values in the examples to your machine configuration.

- The target server must be clean, that is, not running any other WebSphere applications, including any previous version of the Banking Application.

- IBM WebSphere 5.1.1 must be installed and configured. This guide assumes that the WebSphere root directory is: `/usr/WebSphere`

- A database server must be installed and configured; this release supports DB2 and Oracle. This guide assumes that:

    - DB2 is installed at: `/app/IBMdb2/sqllib.`

    - Oracle is installed at: `/opt/oracle/OraHome1.`

- The Java utilities `java`, `javac,` and `jar` must be available at the command line.

- This guide assumes that the CD-ROM drive is mounted at `/mnt/cdrom`; adjust the values in the examples to your machine configuration.

- If interfacing with Siebel Financial Services 7.x is required an installation of Siebel Financial Services 7.x is a prerequisite.

Deploying the Banking Application in a WebSphere Clustered Environment

You install the Banking Application in a clustered environment in the same way as for a nonclustered environment, apart from the steps in this section. Also, for a clustered environment, you use the WebSphere Deployment Manager rather than the WebSphere Administrative Console for configuring the database and JMS components, and for deploying the application.

Requirements for Deploying the Banking Application in a WebSphere Clustered Environment

You must set up and configure the WebSphere clustered environment before deploying the Banking Application.

Consult your vendor's documentation on how to set up and configure a WebSphere clustered environment.

Starting the WebSphere Deployment Manager

You use the Deployment Manager to access each node in the cluster. In a clustered environment you configure the database and JMS components, and deploy the Banking Application using the Deployment Manager.

To start the Deployment Manager

1   Open a Web browser and point it to http://*servername*: *portnumber*/admin, where *servername* is the URL/IP address of the machine on which WebSphere 5.1.1 is running and *portnumber* is the port that the Deployment Manager is listening on.

2   Start the WebSphere Application Server.

The Login screen is displayed.

3   Type any user name in the User ID field. The User ID field is not part of any authentication mechanism but is used to track user changes, therefore you can type any name.

Configuring the Clustered Database and JMS Cluster Settings

You must configure the database and JMS components in the same way as described in Configuring WebSphere Database Settings and Configuring WebSphere JMS Settings with the following exceptions:

■   You must do the configuration using the Deployment Manager rather than the Administrative Console.

■   You must do the configuration on a per-node basis.

Configuring the BankframeResource.properties Cluster Settings

You must configure the BankframeResource.properties file as described in Configuring the BankframeResource.Properties File on page 45, with the following exception:

■   Set the `channel.http.client.url` = property to:

`http://`*WebServer-Hostname*: *WebServer-Port-Number*`/BankFrameMCA/HttpServer`

where *WebServer-Hostname* and *WebServer-Port-Number* are the hostname and port number of the HTTP Server. In most cases, you can leave the port number field blank. Consult the vendor documentation for further information.

Configuring the Application Deployment Cluster Settings

You must configure the settings in the same way as described in Deploying the Banking Application on WebSphere 5.1.1 with the following exceptions:

■   You must do the configuration using the Deployment Manager rather than the Administrative Console.

■   You must select the cluster for all modules on the Map Modules to Application Settings screen.

■ You must select the option Synchronizing changes with Nodes when saving the changes to the repository.

Extracting the SRF Banking Application Files

The Banking Application installation files are located in a JAR file on the Banking Application CD.

NOTE: All instances of *vx.x* in the instructions in this document refer to the version of the software that you are using.

To extract the Banking Application files on Windows

1 Type the command cd /d d:\

2 Type the following command for Oracle or DB2, as applicable, which extracts the files to a new **siebel** directory on **d:** \

■ jar xvf e:\packs\SRFBankingApplication*vx.x*WebSphereOracle.jar

■ jar xvf e:\packs\SRFBankingApplication*vx.x*WebSphereDB2.jar

To extract the Banking Application files on UNIX

1 Type the command cd **db.name** /usr/WebSphere

2 Type the following command for Oracle or DB2, as applicable, which extracts the files to a new **siebel** directory in **/usr/WebSphere:**

■ `jar xvf /mnt/cdrom/packs/SRFBankingApplication`*vx.x*`WebSphereOracle.jar`

■ `jar xvf /mnt/cdrom/packs/SRFBankingApplication`*vx.x*`WebSphereDB2.jar`

Configuring WebSphere Database Settings

This section details the SRF-specific procedures required when configuring WebSphere 5.1.1 for an Oracle or DB2 database server and how to create the required database tables. Some of the sections are specific to Oracle or DB2; ignore the sections that do not apply to your database.

For Oracle only, you must perform the database configuration on a Windows machine, as the database configuration process is only supported on Windows.

Setting up the Oracle Database

The Oracle database for the Siebel RF Banking Application is provided as an Oracle dump file, as some of the tables contain RAW data, which you cannot import using standard SQL insert scripts. To import the data from the dump file into your Oracle database follow the instructions in this section. In these instructions, the existing database is first dropped and then the database is loaded with the contents of the `staging.dmp` file located at `\siebel\database\staging.dmp`. The first time the ant drop or ant load commands are executed you are prompted for a database user name, a database password, and a database instance or db name. This process creates a `database.properties` file at `D:\temp\build_tmp\`. This `database.properties file` determines what database is used by the ant drop and ant load processes. If a different database is used, configure this properties file accordingly.

In the example in this section, it is assumed that the database name, user ID, and password are all set to bankfrm. Adjust these values based on the database name, user ID, and password you chose in the previous section. The following prerequisites apply for importing the Oracle database:

■ The Oracle command line utilities must be available at the command line/console.

■ An Oracle database instance must be created (make a note of the name).

■ An Oracle user for the database must be created (make a note of the user ID and password).

■ An entry for the database must be created in the tnsnames.ora file for the server on which the Banking Application is deployed.

To import the Oracle dump file

1 Type the command `cd /d d:\siebel\database`.

2 To drop the database if it already exists type the command `ant drop`.

3 To load the database type the command `ant load`.

To reset the Oracle database

■ Type the command `ant reset-branches`.

Setting up the DB2 Database

This topic describes how to import the Banking Application DB2 database tables, create the views and sequences, and populate the database tables. You require the following prerequisites for setting up the DB2 database:

■ A DB2 instance running on a server.

■ Physical access to the console of the server running the DB2 instance.

■ The password for the db2admin user on Windows or the db2inst1 user if on UNIX.

■ A database created on this DB2 instance (for example SiebelRF).

■ A new operating system user named BANKFRM with the password BANKFRM and with rights for this user to use the database.

■   The file siebel\database\srf-db2-data.zip copied to the machine hosting the DB2 database. This file contains an export of all the required Banking Application database table data.

■   The file siebel\database\srf-db2.sql copied to the machine hosting the DB2 database. This script is provided to build the Banking Application database tables, views, and sequences.

To create tables, views, and sequences

1   Edit the siebel\database\srf-db2.sql file and replace all occurrences of the following placeholders with the correct values for your environment:

   a   <DB_NAME> - the database name.

   b   <DB_USERNAME> - the user name of the database administrator.

   c   <DB_PASSWORD> - the password of the database administrator.

   d   <TS_PATH> - the path on the file system of the database server where the extra table space required for the Banking Application is stored. (Refer to your DB2 documentation for more information about table spaces).

2   Open a command prompt, navigate to the unzipped folder, and enter the command:

```
db2 –tvf srf-db2.sql
```

To import the database data

1   Unzip the export file siebel\database\srf-db2-data.zip to a folder on the DB2 server machine.

2   Open a command prompt, navigate to the unzipped folder, and enter the command:

```
db2move SiebelRF load import -u BANKFRM -p BANKFRM
```

where *SiebelRF* is the database name.

Creating a Connection to the Database

If the database server is not running on the same machine as the WebSphere server, you must configure a connection to the database on the WebSphere Server. The connection must be named bankfrm. Refer to your database server documentation for information about how to do this.

Starting the WebSphere Application Server

The next step is to start the application server.

To start WebSphere on Windows

1   Navigate to Start > Programs > IBM WebSphere > Application Server v5.1 > First Steps.

2   Click Start the Server.

When `Server server1open for ebusiness` is displayed in the log on the bottom of the panel the server has started successfully.

To start WebSphere on UNIX

3   Navigate to the root of the server installation, this is typically a folder named WebSphere.

4   Change to the bin subdirectory.

5   Type the command:

nohup ./startServer.sh server1 &

Starting the WebSphere Administrative Console

The administrative console is the mechanism by which WebSphere defines and configures its properties.

To start the administrative console

1   Start the WebSphere Application Server.

2   Open a Web browser and point it to http://*servername*:*portnumber*/admin, where *servername* is the URL/IP address of the machine that WebSphere 5.1.1 is running on and *portnumber* is the port the administrative console is listening on.

3   The Login screen is then displayed.

The User ID required is not part of any authentication mechanism and is only used to track user changes, therefore any name is suitable.

Configuring WebSphere's Database Access

You must modify the domain configuration to provide WebSphere with the user credentials associated with the `bankfrm` database. This modification is made to allow WebSphere access to the Banking Application database.

Configuring the Database Driver

The next step is to configure the database driver.

To configure the Oracle or DB2 database driver

1   Log into the WebSphere administrative console as described in Starting the WebSphere Administrative Console.

2   Select Resources > JDBC Providers from the left-side panel.

3   On the screen displayed select the driver name if it is created.

4   If a new driver entry needs to be created, click New, select Oracle JDBC Driver or `DB2 Universal JDBC Provider` from the drop-down list, and click Apply.

The configuration screen is then displayed.

5   Complete the necessary fields for Oracle or DB2, as applicable. Some fields are described in the following table.

Field	Comment
Name	Set the driver name to Oracle JDBC Driver or DB2 Universal JDBC Provider.
Classpath	The classpath to the driver classes is specified by editing the ORACLE_JDBC_DRIVER_PATH variable or the DB2UNIVERSAL_JDBC_DRIVER_PATH variable. This editing is discussed in the next section.
Implementation class name	This name is either: oracle.jdbc.pool.OracleConnectionPoolDataSource or COM.IBM.DB2.JCC.DB2ConnectionPooldataSource

6   Click Apply to save the information, and select Save on the next screen.

Configuring the JDBC Driver Path

When the database driver is declared, you must set the path to the driver classes. You set the path by configuring the ORACLE_JDBC_DRIVER_PATH or DB2UNIVERSAL_JDBC_DRIVER_PATH variable.

To configure the ORACLE or DB2 JDBC driver path

1   Navigate to the Environment > Manage WebSphere Variables > ORACLE_JDBC_DRIVER_PATH or DB2UNIVERSAL_JDBC_DRIVER_PATH screen.

2   Set the Value field to the absolute path of the .zip file containing the driver classes. The .zip file containing the driver classes is usually called classes12.zip for Oracle and db2java.zip for DB2. The path to the zip file is usually D:\oracle\ora\jdbc\lib for Oracle and D:\SQLLIB\java for DB2.

Setting Up a Data Source

After the database driver is declared and a path associated with it, you must set up a data source.

To set up an Oracle or DB2 data source

1   Navigate to the Resources > JDBC Providers > Oracle JDBC Driver/DB2 Universal JDBC Provider > Data Sources (Version 4) screen.

2   Click New to create a new data source. The configuration screen is then displayed.

3   Complete the necessary fields.

Some fields are described in the following table.

Field	Comment
Name	Type bankfrm.
JNDI Name	Type bankfrm.
Database Name	Type bankfrm.
Default User ID	Type bankfrm.
Default Password	Type bankfrm.

4   Navigate to the Custom Properties Screen.

5   Complete the necessary fields.

Some fields are described in the following table.

Field	Comment
serverName	Type the IP Address of the database server machine.
portNumber	Type the TCP/IP port number where the DRDA server resides.

6   Click the URL field on the Custom Properties screen to configure the JDBC URL for Oracle. The URL value is of the format `jdbc:oracle:thin:@hostname:portnumber:SID`, for example, `jdbc:oracle:thin:@database:1521:bankfrm`.

Configuring WebSphere JMS Settings

You must configure JMS settings for:

■   Topic connection factories

■   Destinations

■   Listener ports

To configure WebSphere JMS Topic Connection Factories

1   Navigate to the Resources > WebSphere JMS Provider > WebSphere Topic Connection Factories > New > Configuration screen.

2   Complete the necessary fields. Some fields are described in the following table.

Field	Comment
Name	Type `exampleTopicConnectionFactory`.
JNDI Name	Type `eontec/jms/TopicConnectionFactory`.
Port	Type `Queued`.

3   Set the XA field to enabled at the bottom of the screen, and save the changes.

To configure WebSphere Topic Destinations

1   Navigate to the Resource > WebSphere JMS Provider > WebSphere Topic Destinations > New > General Properties screen.

2   Complete the necessary fields. Some fields are described in the following table.

Field	Comment
Name	Type `exampleTopic`.
JNDI Name	Type `eontec/jms/exampleTopic`.
Topic	Type `eontec/jms/exampleTopic`.

3   Save the configuration changes.

To configure the Eontec Listener

1   Navigate to the Servers >Application Servers > server1 > Message Listener Service > Listener Ports > New > Configuration screen.

2   Complete the necessary fields. Some fields are described in the following table.

Field	Comment
Name	Type `eontecListener`.
Initial State	Type `started`.
Connection Factory JNDI Name	Type `eontec/jms/TopicConnectionFactory`.
Destination JNDI Name	Type `eontec/jms/exampleTopic`.

3   Save the configuration changes.

Configuring Pass by Reference

You must configure WebSphere 5.1.1 to pass EJB instances by reference.

To configure pass by reference

1   Navigate to the Servers > Application Servers > server1 > ORB Service screen.

2   Enable the Pass by reference field.

3   Save the configuration changes.

Configuring the BankframeResource.Properties File

You must configure the file BankframeResource.properties for local settings before the banking application is uploaded onto the server. The BankframeResource.properties file is located in `d:\siebel\deploy\siebel.ear`.

To edit the file BankframeResource.properties

1   Extract the file BankframeResource.properties from `Siebel.ear` and open it in a text editor.

2   Search for the string `ejb.server`= and replace all text to the right of this string with `iiop://localhost` where localhost is the machine name, complete with domain suffixes.

3   Search for the `string ejb.jndiSyntax`= and Replace any text to the right of this string with the value `1.1`.

4   Search for the `string ejb.jndiPrefix`= and make sure that there is no value to the right of this text; it must be blank.

5   Search for the string `channel.http.client.url`= and replace all text to the right of this string with http://localhost:9080/BankFrameMCA/HttpServer where localhost is the machine name, complete with domain suffixes.

6   Remove or add the comment (#) in the sequence for Oracle/DB2 in the Sequence Generation section, as appropriate.

The following example shows the appropriate configuration when the host database is DB2.

`# Sequence Generation`

`# Note: keys for sql statements need only be set if you do not wish to use the`

` default`

`# (as seen below)`

`###########################`

`#mca.services.sequences.factoryClass=com.bankframe.services.sequences.OracleSequenceGeneratorFactoryImpl`

`mca.services.sequences.factoryClass=com.bankframe.services.sequences.DB2SequenceGeneratorFactoryImpl`

`mca.services.sequences.datasource=bankfrm`

`#mca.services.sequences.sql.oracle.nextval=SELECT ?.nextval FROM DUAL`

`mca.services.sequences.sql.db2.nextval=SELECT NEXTVAL FOR ? FROM SYSIBM.SYSVERSIONS`

7   Remove the comment (#) from the following SSL settings for WebSphere:

`# Websphere`

`#channel.https.ssl.protocol=com.ibm.net.ssl.internal.www.protocol`

`#channel.https.ssl.provider=com.ibm.jsse.JSSEProvider`

8   Save the configuration changes.

9   Right-click on the modified BankframeResource.properties file, and select Add to Zip.

10 Browse to the `siebel.ear` file that BankframeResource.properties was extracted from.

11 Re-add the modified BankframeResource.properties file to the EAR file, maintaining the path structure, that is, `\resources\BankframeResource.properties`.

The EAR file is now ready to install on the server.

Deploying the Banking Application on WebSphere 5.1.1

This section assumes that the procedures for configuring the WebSphere 5.1.1 server and setting up and populating the database are completed and that the WebSphere Web Browser Console is displayed.

To deploy the Banking Application on WebSphere 5.1.1

1   Navigate to the Applications > Enterprise Applications > Install > Preparing for the application installation screen.

2   Click Browse, and select the file siebel.ear on the local machine.

3   Click Next to upload the siebel.ear file from the local machine to the server, whether it is on a remote machine or on the local machine itself.

The AppDeployment Options screen is then displayed.

4   Select Next on each of the following nine screens to accept the default settings. The default settings are preset when the `siebel.ear` file is uploaded.

5   Click Finish and Save to deploy the EAR file.

6   Navigate to the Applications > Enterprise Applications screen.

A list of installed applications is displayed and the Siebel application has a red x to the right of it, signifying that the application is installed but not running.

7   Click the check box to the left of the Siebel application, click Start, and then save the changes.

8   Navigate to the Applications > Enterprise Applications screen. A list of installed applications is displayed and the Siebel application has a green arrow to the right of it, signifying that the application is installed and running.

5 WebSphere 6.1 Installation Process

This chapter describes how to install the Banking Application on the WebSphere platform. It contains the following topics:

- Requirements for Installing the Banking Application on Windows
- Requirements for Installing the Banking Application on UNIX
- Deploying the Banking Application in a WebSphere Clustered Environment
- Extracting the SRF Banking Application Files
- Configuring WebSphere Database Settings
- Configuring WebSphere JMS Settings
- Configuring Pass by Reference
- Configuring the BankframeResource.Properties File
- Deploying the Banking Application on WebSphere 6.1

NOTE: Refer to the Siebel Retail Finance System Requirements and Supported Platforms document on the Siebel SupportWeb Web site (http://supportweb.siebel.com) for information regarding the supported environments, including the supported database and application server versions, and supported versions of third- party software. Your Siebel TAM will have provided you with a SupportWeb logon.

Requirements for Installing the Banking Application on Windows

This section outlines the installation prerequisites when installing the Banking Application on Windows. This guide assumes the installation locations in the following list; adjust the values in the examples to your machine configuration.

- The target server must be clean, that is, not running any other WebSphere applications, including any previous version of the Banking Application.
- IBM WebSphere 6.1 must be installed and configured. This guide assumes that the WebSphere root directory is: `D:\Program Files\WebSphere.`
- A database server must be installed and configured; this release supports both DB2 and Oracle. This guide assumes that:
  - DB2 is installed at: `D:\SQLLIB.`
  - Oracle is installed at: `D:\Oracle.`
- The Java utilities `java`, `javac`, and `jar` must be available at the command line.
- This guide assumes that the CD-ROM drive is attached to drive `E:\`; adjust the values in the examples to your machine configuration.

■   If interfacing with Siebel Financial Services 7.x is required an installation of Siebel Financial Services 7.x is a prerequisite.

Requirements for Installing the Banking Application on UNIX

This section outlines the installation prerequisites when installing the Banking Application on UNIX. This guide assumes the installation locations listed below; adjust the values in the examples to your machine configuration.

■   The target server must be clean, that is, not running any other WebSphere applications, including any previous version of the Banking Application.

■   IBM WebSphere 6.1 must be installed and configured. This guide assumes that the WebSphere root directory is: `/usr/WebSphere`

■   A database server must be installed and configured; this release supports DB2 and Oracle. This guide assumes that:

   ■   DB2 is installed at: `/app/IBMdb2/sqllib.`

   ■   Oracle is installed at: `/opt/oracle/OraHome1.`

■   The Java utilities `java`, `javac`, and `jar` must be available at the command line.

■   This guide assumes that the CD-ROM drive is mounted at `/mnt/cdrom`; adjust the values in the examples to your machine configuration.

■   If interfacing with Siebel Financial Services 7.x is required an installation of Siebel Financial Services 7.x is a prerequisite.

Deploying the Banking Application in a WebSphere Clustered Environment

You install the Banking Application in a clustered environment in the same way as for a nonclustered environment, apart from the steps in this section. Also, for a clustered environment, you use the WebSphere Deployment Manager rather than the WebSphere Administrative Console for configuring the database and JMS components, and for deploying the application.

Requirements for Deploying the Banking Application in a WebSphere Clustered Environment

You must set up and configure the WebSphere clustered environment before deploying the Banking Application.

Consult your vendor's documentation on how to set up and configure a WebSphere clustered environment.

Starting the WebSphere Deployment Manager

You use the Deployment Manager to access each node in the cluster. In a clustered environment you configure the database and JMS components, and deploy the Banking Application using the Deployment Manager.

To start the Deployment Manager

1 Open a Web browser and point it to http://*servername*: *portnumber*/admin, where *servername* is the URL/IP address of the machine on which WebSphere 6.1 is running and *portnumber* is the port that the Deployment Manager is listening on.

2 Start the WebSphere Application Server.

The Login screen is displayed.

3 Type any user name in the User ID field. The User ID field is not part of any authentication mechanism but is used to track user changes, therefore you can type any name.

Configuring the Clustered Database and JMS Cluster Settings

You must configure the database and JMS components in the same way as described in Configuring WebSphere Database Settings and Configuring WebSphere JMS Settings with the following exceptions:

■ You must do the configuration using the Deployment Manager rather than the Administrative Console.

■ You must do the configuration on a per-node basis.

Configuring the BankframeResource.properties Cluster Settings

You must configure the BankframeResource.properties file as described in Configuring the BankframeResource.Properties File on page 45, with the following exception:

■ Set the `channel.http.client.url` = property to:

$http://WebServer\text{-}Hostname: WebServer\text{-}Port\text{-}Number/BankFrameMCA/HttpServer$

where *WebServer-Hostname* and *WebServer-Port-Number* are the hostname and port number of the HTTP Server. In most cases, you can leave the port number field blank. Consult the vendor documentation for further information.

Configuring the Application Deployment Cluster Settings

You must configure the settings in the same way as described in Deploying the Banking Application on WebSphere 6.1 with the following exceptions:

■ You must do the configuration using the Deployment Manager rather than the Administrative Console.

■ You must select the cluster for all modules on the Map Modules to Application Settings screen.

■ You must select the option Synchronizing changes with Nodes when saving the changes to the repository.

Extracting the SRF Banking Application Files

The Banking Application installation files are located in a JAR file on the Banking Application CD.

NOTE: All instances of *vx.x* in the instructions in this document refer to the version of the software that you are using.

To extract the Banking Application files on Windows

1 Type the command cd /d d:\

2 Type the following command for Oracle or DB2, as applicable, which extracts the files to a new **siebel** directory on **d:** \

■ jar xvf e:\packs\SRFBankingApplication*vx.x*WebSphereOracle.jar

■ jar xvf e:\packs\SRFBankingApplication*vx.x*WebSphereDB2.jar

To extract the Banking Application files on UNIX

1 Type the command cd **db.name** /usr/WebSphere

2 Type the following command for Oracle or DB2, as applicable, which extracts the files to a new **siebel** directory in **/usr/WebSphere:**

■ `jar xvf /mnt/cdrom/packs/SRFBankingApplication`*vx.x*`WebSphereOracle.jar`

■ `jar xvf /mnt/cdrom/packs/SRFBankingApplication`*vx.x*`WebSphereDB2.jar`

Configuring WebSphere Database Settings

This section details the SRF-specific procedures required when configuring WebSphere 6.1 for an Oracle or DB2 database server and how to create the required database tables. Some of the sections are specific to Oracle or DB2; ignore the sections that do not apply to your database.

For Oracle only, you must perform the database configuration on a Windows machine, as the database configuration process is only supported on Windows.

Setting up the Oracle Database

The Oracle database for the Siebel RF Banking Application is provided as an Oracle dump file, as some of the tables contain RAW data, which you cannot import using standard SQL insert scripts. To import the data from the dump file into your Oracle database follow the instructions in this section. In these instructions, the existing database is first dropped and then the database is loaded with the contents of the `staging.dmp` file located at `\siebel\database\staging.dmp`. The first time the ant drop or ant load commands are executed you are prompted for a database user name, a database password, and a database instance or db name. This process creates a `database.properties` file at `D:\temp\build_tmp\`. This `database.properties file` determines what database is used by the ant drop and ant load processes. If a different database is used, configure this properties file accordingly.

In the example in this section, it is assumed that the database name, user ID, and password are all set to bankfrm. Adjust these values based on the database name, user ID, and password you chose in the previous section. The following prerequisites apply for importing the Oracle database:

- The Oracle command line utilities must be available at the command line/console.

- An Oracle database instance must be created (make a note of the name).

- An Oracle user for the database must be created (make a note of the user ID and password).

- An entry for the database must be created in the tnsnames.ora file for the server on which the Banking Application is deployed.

To import the Oracle dump file

1   Type the command `cd /d d:\siebel\database`.

2   To drop the database if it already exists type the command `ant drop`.

3   To load the database type the command `ant load`.

To reset the Oracle database

- Type the command `ant reset-branches`.

Setting up the DB2 Database

This topic describes how to import the Banking Application DB2 database tables, create the views and sequences, and populate the database tables. You require the following prerequisites for setting up the DB2 database:

- A DB2 instance running on a server.

- Physical access to the console of the server running the DB2 instance.

- The password for the db2admin user on Windows or the db2inst1 user if on UNIX.

- A database created on this DB2 instance (for example SiebelRF).

- A new operating system user named BANKFRM with the password BANKFRM and with rights for this user to use the database.

■ The file siebel\database\srf-db2-data.zip copied to the machine hosting the DB2 database. This file contains an export of all the required Banking Application database table data.

■ The file siebel\database\srf-db2.sql copied to the machine hosting the DB2 database. This script is provided to build the Banking Application database tables, views, and sequences.

To create tables, views, and sequences

1 Edit the siebel\database\srf-db2.sql file and replace all occurrences of the following placeholders with the correct values for your environment:

  e <DB_NAME> - the database name.

  f <DB_USERNAME> - the user name of the database administrator.

  g <DB_PASSWORD> - the password of the database administrator.

  h <TS_PATH> - the path on the file system of the database server where the extra table space required for the Banking Application is stored. (Refer to your DB2 documentation for more information about table spaces).

2 Open a command prompt, navigate to the unzipped folder, and enter the command:

```
db2 –tvf srf-db2.sql
```

To import the database data

1 Unzip the export file siebel\database\srf-db2-data.zip to a folder on the DB2 server machine.

2 Open a command prompt, navigate to the unzipped folder, and enter the command:

```
db2move SiebelRF load import -u BANKFRM -p BANKFRM
```

where *SiebelRF* is the database name.

Creating a Connection to the Database

If the database server is not running on the same machine as the WebSphere server, you must configure a connection to the database on the WebSphere Server. The connection must be named bankfrm. Refer to your database server documentation for information about how to do this.

Starting the WebSphere Application Server

The next step is to start the application server.

To start WebSphere on Windows

3 Navigate to Start > Programs > IBM WebSphere > Application Server v6.1 > First Steps.

4 Click Start the Server.

When `Server server1open for ebusiness` is displayed in the log on the bottom of the panel the server has started successfully.

To start WebSphere on UNIX

1   Navigate to the root of the server installation, this is typically a folder named WebSphere.

2   Change to the bin subdirectory.

3   Type the command:

nohup ./startServer.sh server1 &

Starting the WebSphere Administrative Console

The administrative console is the mechanism by which WebSphere defines and configures its properties.

To start the administrative console

1   Start the WebSphere Application Server.

2   Open a Web browser and point it to http://*servername*:*portnumber*/admin, where *servername* is the URL/IP address of the machine that WebSphere 6.1 is running on and *portnumber* is the port the administrative console is listening on.

3   The Login screen is then displayed.

The User ID required is not part of any authentication mechanism and is only used to track user changes, therefore any name is suitable.

Configuring WebSphere's Database Access

You must modify the domain configuration to provide WebSphere with the user credentials associated with the `bankfrm` database. This modification is made to allow WebSphere access to the Banking Application database.

Configuring the Database Driver

The next step is to configure the database driver.

To configure the Oracle or DB2 database driver

1   Log into the WebSphere administrative console as described in Starting the WebSphere Administrative Console.

2   Select Resources > JDBC Providers from the left-side panel.

3   On the screen displayed select the driver name if it is created.

4   If a new driver entry needs to be created, click New, select Oracle JDBC Driver or `DB2 Universal JDBC Provider` from the drop-down list, and click Apply.

The configuration screen is then displayed.

5   Complete the necessary fields for Oracle or DB2, as applicable. Some fields are described in the following table.

Field	Comment
Name	Set the driver name to Oracle JDBC Driver or `DB2 Universal JDBC Provider.`
Classpath	The classpath to the driver classes is specified by editing the `ORACLE_JDBC_DRIVER_PATH` variable or the `DB2UNIVERSAL_JDBC_DRIVER_PATH` variable. This editing is discussed in the next section.
Implementation class name	This name is either: oracle.jdbc.pool.OracleConnectionPoolDataSource or `COM.IBM.DB2.JCC.DB2ConnectionPooldataSource`

6   Click Apply to save the information, and select Save on the next screen.

Configuring the JDBC Driver Path

When the database driver is declared, you must set the path to the driver classes. You set the path by configuring the ORACLE_JDBC_DRIVER_PATH or DB2UNIVERSAL_JDBC_DRIVER_PATH variable.

To configure the ORACLE or DB2 JDBC driver path

1   Navigate to the Environment > Manage WebSphere Variables > ORACLE_JDBC_DRIVER_PATH or DB2UNIVERSAL_JDBC_DRIVER_PATH screen.

2   Set the Value field to the absolute path of the .zip file containing the driver classes. The .zip file containing the driver classes is usually called classes12.zip for Oracle and `db2java.zip` for DB2. The path to the zip file is usually $\mathbf{D:\oracle\ora\jdbc\lib}$ for Oracle and $\mathbf{D:\SQLLIB\java}$ for DB2.

Setting Up a Data Source

After the database driver is declared and a path associated with it, you must set up a data source.

To set up an Oracle or DB2 data source

1   Navigate to the Resources > JDBC Providers > Oracle JDBC Driver/DB2 Universal JDBC Provider > Data Sources (Version 4) screen.

2   Click New to create a new data source. The configuration screen is then displayed.

3   Complete the necessary fields.

Some fields are described in the following table.

Field	Comment
Name	Type bankfrm.
JNDI Name	Type bankfrm.
Database Name	Type bankfrm.
Default User ID	Type bankfrm.
Default Password	Type bankfrm.

4   Navigate to the Custom Properties Screen.

5   Complete the necessary fields.

Some fields are described in the following table.

Field	Comment
serverName	Type the IP Address of the database server machine.
portNumber	Type the TCP/IP port number where the DRDA server resides.

6   Click the URL field on the Custom Properties screen to configure the JDBC URL for Oracle. The URL value is of the format `jdbc:oracle:thin:@hostname:portnumber:SID`, for example, `jdbc:oracle:thin:@database:1521:bankfrm`.

Configuring WebSphere JMS Settings

You must configure JMS settings for:

- Topic connection factories
- Destinations
- Listener ports

To configure WebSphere JMS Topic Connection Factories

1   Navigate to the Resources > WebSphere JMS Provider > WebSphere Topic Connection Factories > New > Configuration screen.

2   Complete the necessary fields. Some fields are described in the following table.

Field	Comment
Name	Type `exampleTopicConnectionFactory`.
JNDI Name	Type `eontec/jms/TopicConnectionFactory`.
Port	Type `Queued`.

3   Set the XA field to enabled at the bottom of the screen, and save the changes.

To configure WebSphere Topic Destinations

1   Navigate to the Resource > WebSphere JMS Provider > WebSphere Topic Destinations > New > General Properties screen.

2   Complete the necessary fields. Some fields are described in the following table.

Field	Comment
Name	Type `exampleTopic`.
JNDI Name	Type `eontec/jms/exampleTopic`.
Topic	Type `eontec/jms/exampleTopic`.

3   Save the configuration changes.

To configure the Eontec Listener

1   Navigate to the Servers >Application Servers > server1 > Message Listener Service > Listener Ports > New > Configuration screen.

2   Complete the necessary fields. Some fields are described in the following table.

Field	Comment
Name	Type `eontecListener`.
Initial State	Type `started`.
Connection Factory JNDI Name	Type `eontec/jms/TopicConnectionFactory`.
Destination JNDI Name	Type `eontec/jms/exampleTopic`.

3   Save the configuration changes.

Configuring Pass by Reference

You must configure WebSphere 6.1 to pass EJB instances by reference.

To configure pass by reference

1   Navigate to the Servers > Application Servers > server1 > ORB Service screen.

2   Enable the Pass by reference field.

3   Save the configuration changes.

Configuring the BankframeResource.Properties File

You must configure the file BankframeResource.properties for local settings before the banking application is uploaded onto the server. The BankframeResource.properties file is located in `d:\siebel\deploy\siebel.ear`.

To edit the file BankframeResource.properties

1   Extract the file BankframeResource.properties from `Siebel.ear` and open it in a text editor.

2   Search for the string `ejb.server=` and replace all text to the right of this string with `iiop://localhost` where localhost is the machine name, complete with domain suffixes.

3   Search for the `string ejb.jndiSyntax=` and Replace any text to the right of this string with the value `1.1`.

4   Search for the `string ejb.jndiPrefix=` and make sure that there is no value to the right of this text; it must be blank.

5   Search for the string `channel.http.client.url=` and replace all text to the right of this string with http://localhost:9080/BankFrameMCA/HttpServer where localhost is the machine name, complete with domain suffixes.

6   Remove or add the comment (#) in the sequence for Oracle/DB2 in the Sequence Generation section, as appropriate.

The following example shows the appropriate configuration when the host database is DB2.

`# Sequence Generation`

`# Note: keys for sql statements need only be set if you do not wish to use the`

`  default`

`# (as seen below)`

`###########################`

`#mca.services.sequences.factoryClass=com.bankframe.services.sequences.OracleSequenceGeneratorFactoryImpl`

`mca.services.sequences.factoryClass=com.bankframe.services.sequences.DB2SequenceGeneratorFactoryImpl`

`mca.services.sequences.datasource=bankfrm`

`#mca.services.sequences.sql.oracle.nextval=SELECT ?.nextval FROM DUAL`

`mca.services.sequences.sql.db2.nextval=SELECT NEXTVAL FOR ? FROM SYSIBM.SYSVERSIONS`

7   Remove the comment (#) from the following SSL settings for WebSphere:

`# Websphere`

`#channel.https.ssl.protocol=com.ibm.net.ssl.internal.www.protocol`

`#channel.https.ssl.provider=com.ibm.jsse.JSSEProvider`

8 Save the configuration changes.

9 Right-click on the modified BankframeResource.properties file, and select Add to Zip.

10 Browse to the `siebel.ear` file that BankframeResource.properties was extracted from.

11 Re-add the modified BankframeResource.properties file to the EAR file, maintaining the path structure, that is, \resources**BankframeResource.properties**.

The EAR file is now ready to install on the server.

Deploying the Banking Application on WebSphere 6.1

This section assumes that the procedures for configuring the WebSphere 6.1 server and setting up and populating the database are completed and that the WebSphere Web Browser Console is displayed.

To deploy the Banking Application on WebSphere 6.1

1 Navigate to the Applications > Enterprise Applications > Install > Preparing for the application installation screen.

2 Click Browse, and select the file siebel.ear on the local machine.

3 Click Next to upload the siebel.ear file from the local machine to the server, whether it is on a remote machine or on the local machine itself.

The AppDeployment Options screen is then displayed.

4 Select "Detailed - Show all installation options and parameters" option and click Next.

5 Select Next in the following two windows to accept the default settings.

6 In the next window, called "Provide options to perform the EJB deploy", change the default Database type, DB2UDB_V82 by selecting blank database from the drop down.

7 Select Next for all remaining windows. The default settings are preset when the `siebel.ear` file is uploaded.

8 Click Finish and Save to deploy the EAR file.

9 Navigate to the Applications > Enterprise Applications screen.

A list of installed applications is displayed and the Siebel application has a red x to the right of it, signifying that the application is installed but not running.

10 Click the check box to the left of the Siebel application, click Start, and then save the changes.

11 Navigate to the Applications > Enterprise Applications screen. A list of installed applications is displayed and the Siebel application has a green arrow to the right of it, signifying that the application is installed and running.

6 Installing Third Party Software

This chapter describes how to install the third party software that is required to enable peripheral devices and PDF form generation for the Banking Application.

It includes the following topics:

- Installing the Java Communications API
- Enabling the Banking Application to be Rebuilt with Peripheral Support
- Enabling PDF Form Generation

Installing the Java Communications API

Support for peripheral devices connected to RS232 serial ports and IEEE 1284 parallel ports is provided through the Java Communications API. You must download and install this third party software, if peripheral devices are to be used in your Banking Application installation.

To install the Java Communications API

1   Open the URL http://java.sun.com/products/javacomm/downloads/index.html to download the Java Communications API.

2   Select Version 2.0 for Microsoft Windows and Solaris/x86 package.

3   Accept the license agreement prompt, and select the javacomm20-win32.zip URL.

4   Save the zip file to a temporary folder, for example, c:\temp.

5   Extract the contents of javacomm20-win32.zip to the temporary folder.

Installing the Java Communications API Extension in the JDK/JRE

Because the Java Communications API relies on a native library, you must install it into your JDK or JRE so that the JDK can load the native library at run time.

If you are installing the extension on a JDK, note that you must install all files in the jre subfolder of the JDK.

To install the Java Communications API extension

1   Place the win32com.dll in the *jre*\bin directory.

    where jre is the root directory of your JRE installation.

2   Place the comm.jar file in the *jre*\lib\ext directory.

3   Place the javax.comm.properties in the *jre*\lib directory.

Enabling the Banking Application to be Rebuilt with Peripheral Support

To enable the Banking Application to be rebuilt with peripheral support, the comm.jar file must be in its expected location.

The 3rdParty project as shipped expects the comm.jar file to be at the \siebel\3rdParty\lib\comm\Win32\comm.jar location.

You must therefore copy the comm.jar file from the c:\temp\commapi folder to the \siebel\3rdParty\lib\comm\Win32 folder.

Enabling PDF Form Generation

The currency transaction report (CTR) functionality supports the creation of a PDF form of a CTR. To enable this functionality, you must download and install the Adobe Forms Data Format (FDF) Toolkit for Java to the expected location.

To install the FDF Toolkit for Java

1   Open the URL: http://partners.adobe.com/public/developer/acrobat/fdf/topic.html.

2   Complete the application form to request the software.

3   When your application is accepted, follow the instructions on the Web site for downloading and installing the software.

The FDF Toolkit for Java library must be present to rebuild the BranchClient and BranchAdministrator projects. You must therefore copy the jFdfTk.jar file to the \siebel\3rdParty\lib\fdf folder before you rebuild the BranchClient and BranchAdministrator projects.

7 Starting the Banking Applications

This chapter describes how to start the Banking Application. It contains the following topics:

- Starting the Branch Teller Application
- Starting the Internet Banking Application

Starting the Branch Teller Application

Before you start the Banking Application the following conditions are required:

- Sun J2SE 1.4.2 JRE is installed on the Branch Teller client machine or machines.
- The Banking Application and supports supplied in this release are deployed as per the instructions in this document.
- The Application Server and the application (siebel.ear) have started successfully.

Starting the Application

Use the following URLs to access the Branch Teller client and Branch Teller Administration Console functionality contained in the Banking Application:

- http://*servername*:*port*/branchteller
- http://*servername:port*/branchadmin

where *servername* is the name of the server to which the Banking Application is deployed, and *port* is the port that the Application Server is listening on (typically 9080 for WebSphere and 7001 for WebLogic).

Running MCA Services Administration Services

Use the following URL to access the MCA Services administration functionality contained in the Banking Application:

http://*servername*:*port*/BankFrameMCA/ServiceServlet

where *servername* is the name of the on which server the Banking Application is deployed and *port* is the port that the application server is listening on (typically 9080 for WebSphere and 7001 for WebLogic).

Sample Branch Teller Logons

Logons contained in the sample data contained in the *vx.x* release of the Banking Application include:

Table 1. Sample Branch Teller Logons

Name	Logon Name	Password	Branch	User Type	User ID	Privileges/ Actor Group
Peter Foley	pfoley	pfoley	9000000000	Administrator	000000000002001	Central Administrator
Karen Saunders	ksaunders	ksaunders	9000000001	Administrator	000000000000037	Central Administrator, Entitlements Administrator
Karen Marino	kmarino	kmarino	9000000001	Teller	000000000001529	Teller
Susan Drew	sdrew	sdrew	9000000001	Teller	000000000001530	Senior Teller
David Reid	dreid	dreid	9000000001	Supervisor	000000000001527	Supervisor

Starting the Internet Banking Application

This section assumes that the Banking Application and supports supplied in this release are deployed, and that the application server and the application (siebel.ear) are started successfully.

Starting the Application

You can use these URLs to access the Internet Banking application and the Online Registration:

- `http://servername:port/internetbanking/StateMachine`
- `http://servername:port/onlinereg/StateMachine`

where *servername* is the name of the server where Banking Application is deployed and *port* is the port on which the application server is listening.

Sample Internet Banking/Entitlements Account Data

The following sections describe account data shipped with the Banking Application.

Administrator Data

The following administrator accounts are set up in the default data that is shipped with the Banking Application:

Table 2. Administrator Account Details

Name	Logon Name	Password	Branch	User Type	User ID	Privileges/ Actor Groups
Peter Foley	pfoley	pfoley	9000000000	Administrator	000000000002001	Central Administrator
Karen Saunders	ksaunders	ksaunders	9000000001	Administrator	000000000000037	Central Administrator

Unregistered Customer Data

You can use the following sample data shipped with the Banking Application to register customers for the Internet Banking service:

Table 3. Unregistered Customer Data

Owner ID	First Name	Surname	Branch Code	Account No.	Services Value
100000000000222	Felix	Aaron	9000000002	5000000127	ATM, 4600 0000 0000 0122, PIN 4575
100000000000226	Diana	Abbot	9000000001	5000000120	ATM, 4600 0000 0000 0115, PIN 8944
100000000000229	George	Abby	9000000001	5000000122	ATM, 4600 0000 0000 0118, PIN 7040 Check Guarantee, 4600 0000 0000 0119, PIN 7090 Debit Card, 4600 0000 0000 0120, PIN 4766

8 Installing the Branch Offline Server

This chapter describes how to install the Branch Offline Server. It contains the following topics:

- Prerequisites
- Extracting the Branch Offline Server Files
- Configuring the Branch Offline Server
- Configuring run.bat and Starting the Branch Offline Server
- Running Branch Offline

Prerequisites

Before you install the Branch Offline Server make sure that:

- The Branch Teller Application (Central Server) is running.
- The Windows platform machine, where Branch Offline Server is to be installed, has its hard drive partitioned as c: (primary hard drive) and d: (logical hard drive) in the extended partition.

Extracting the Branch Offline Server Files

The Branch Offline Server software is located on the Banking Application CD for each application server/database platform in a file called **BranchServer.zip.**

You must extract BranchServer.zip to the root of your d:\ drive.

Configuring the Branch Offline Server

You must edit the **BranchServer\resources\BankframeResource.properties file** to point to your central application server.

To configure the Branch Offline Server

1 Open d:\Branchserver\resources\BankframeResource.properties with a text editor.

2 Find the section of the file shown in the following example, and update the channel.http.client.url entry to point to your online application server:

```
####################################################
# URL of the Channel Server to communicate with
# Possible values are:
# http://localhost:7001/HttpServer (WebLogic 5.1)
```

```
# http://localhost/HttpServer (WebSphere 3.5)

# http://localhost:7001/BankFrameMCA/HttpServer (WebLogic6.1)

# http://localhost/BankFrameMCA/HttpServer (WebSphere 4.0) 1

channel.http.client.url=http://
YourCentralServer'sIPAddress:YourCentralServer'sPortNumber/BankFrameMCA/HttpServer
```

3 Save the changes to BankFrameResource.properties.

Configuring run.bat and Starting the Branch Offline Server

To start the Branch Offline Server, you must first configure the run.bat file.

To start the Branch Offline Server

1 Edit the `D:\BranchServer\run.bat` file to point to your JAVA_HOME. For example:

`set JAVA_HOME=D:\java\j2sdk1.4.2_04`

2 Save `run.bat`.

3 Change directory to the location of your run.bat, typically `d:\BranchServer`.

4 Enter the following command:

`Run dba sql`

where dba and sql are the username and password for the Sybase database. The Offline server database and Offline Branch Server are started.

Running Branch Offline

You must start the components of Branch Teller in this order:

1 Start the Central Application Server.

2 Start the Branch Offline Server.

3 Start the Branch Teller Client Application.

You must complete the Branch Start of Day operation for your supervisor and teller users online. Completing this operation updates `the offline database` with all the required data, including transaction fees data.

---

[1] For WebSphere you only require the following change: *channel.http.client.url=http://
YourCentralServer'sIPAddress/BankFrameMCA/HttpServer.*

Branch Start of Day need only be carried out once online. After that it can be done offline. After each user has logged in online once, he or she can then log on in offline mode.

The Branch runs in offline mode under the following circumstances:

- The application server is down.

- The connection between the application server and the branch offline server is severed.

- The `OfflineTest` file is modified. This modification enables a switch to offline mode even when the application server is running. The location of the `OfflineTest` file is specified by the value of the—`Doffline.helper.test.file` setting in the offline `run.bat` file, the default location is `C:\`. Modifying the online/offline setting enables testing to switch between online and offline mode without stopping the application server.

  The `OfflineTest` file has two entries. You can switch between online and offline modes by swapping the comment character (#) at the start of the applicable line. For example, the following configuration results in an offline response:

  - `com.siebel.rf.offline.ei.channel.client.AlwaysOfflineClient`

  - `#com.siebel.rf.webstart.client.WebStartChannelClient`

9 Siebel Retail Finance Integration with Siebel 7.x

This chapter describes the integration of Branch Teller with the Siebel 7.x application. It contains the following topics:

- Integration Prerequisites
- Configuring the Database for Integration with Siebel Financial Services

Integration Prerequisites

If the integration of Branch Teller with the Siebel Financial Services 7.x is required then an installation and configuration of the Siebel Financial Services 7.x application is required. Refer to the Siebel documentation available on Siebel SupportWeb for more information on this configuration.

Some software resources are required for this configuration and these configuration resources are located in the SiebelTCRCIntegrationConfigurationResources.jar file on the Banking Application CD for each application server/database platform supported in this release.

Configuring the Database for Integration with Siebel Financial Services

For full integration with Siebel Financial Services 7.x, you must configure the Banking Application with the location of the machine on which the Siebel Financial Services 7.x Application Server is hosted. You do this by modifying the CONNECTOR_PROPERTIES column in the DESTINATION table.

The Siebel Financial Services integration related DESTINATION_IDs in the DESTINATION table correspond to Customer Relationship Console (CRC) processes as shown in Table 6.

Table 4. Mapping of the DESTINATION_ID Column in the DESTINATION Table to CRC Processes

DESTINATION_ID Column	CRC Process Description
CRC001	Get the initial contact Summary Info from the Siebel application. This information includes Customer Summary, Offers, and Alerts.
CRC002	Get the list of offers for a contact from the Siebel application. This information is used to pull the Offers for a contact after a reject or accept execution on a displayed offer.
CRC003	Get the Referrals records from the Siebel application.
CRC004	Create a Referral record in the Siebel

DESTINATION_ID Column	CRC Process Description
	application.
CRC005	Create a Response record in the Siebel application.

In the CONNECTOR_PROPERTIES column in the DESTINATION table, you must replace the string represented by {SIEBEL_HOST} in Table 7 with the IP Address or machine name (complete with domain suffixes) of the machine hosting the Siebel Financial Services 7.x Application Server to be integrated. An example of a suitable SQL INSERT statement is as follows:

```
INSERT INTO DESTINATION (DESTINATION_ID, CONNECTOR_FACTORY_CLASSNAME,
CONNECTOR_PROPERTIES, HOST_STATUS, IS_EAB ) VALUES ( 'CRC001',
'com.bankframe.ei.txnhandler.connector.http.HTTPConnectionFactory', 'URL_STRING-
http://lhurleydp4/eai_enu/start.swe?SWEExtCmd=Execute&SWEExtSource=TCRCQueryContac
t&UserName=${CRCUserName}&Password=${CRCPassword}', 'ON_LINE', 'False');
```

Table 5. Values for CRC Integration in DESTINATION Table

DESTINATION_ID	CONNECTOR_FACTORY_CLASSNAME	CONNECTOR_PROPERTIES	HOST_STATUS	IS_EAB
CRC001	com.bankframe.ei.txnhandler.connector.http.HTTPConnectionFactory	URL_STRING-http://{SIEBEL_HOST}/eai_enu/start.swe?SWEExtCmd=Execute&SWEExtSource=TCRCQueryContact&UserName=${CRCUserName}&Password=${CRCPassword}	ON_LINE	False
CRC002	com.bankframe.ei.txnhandler.connector.http.HTTPConnectionFactory	URL_STRING-http://{SIEBEL_HOST}/eai_enu/start.swe?SWEExtCmd=Execute&SWEExtSource=TCRCQueryOffer&UserName=${CRCUserName}&Password=${CRCPassword}	ON_LINE	False
CRC003	com.bankframe.ei.txnhandler.connector.http.HTTPConnectionFactory	URL_STRING-http://{SIEBEL_HOST}/eai_enu/start.swe?SWEExtCmd=Execute&SWEExtSource=TCRCQueryOpportunity&UserName=${CRCUserName}&Password=${CRCPassword}	ON_LINE	False
CRC004	com.bankframe.ei.txnhandler.connector.http.HTTPConnectionFactory	URL_STRING-http://{SIEBEL_HOST}/eai_enu/start.swe?SWEExtCmd=Execute&SWEExtSource=TCRCInsertReferral&UserName=${CRCUserName}&Password=${CRCPassword}	ON_LINE	False
CRC005	com.bankframe.ei.txnhandler.connector.http.HTTPConnectionF	URL_STRING-http://{SIEBEL_HOST}/eai_enu/start.swe?SWEExtCmd=Execute&SWEExtSource=TCRCInser	ON_LINE	False

DESTINA TION_ID	CONNECTOR_FACT ORY _CLASSNAME	CONNECTOR_ PROPERTIES	HOST_ STATUS	IS_EAB
	actory	tResponse&UserName=${CRCUserName}&Password=${CRCPassword}		

At run time, the strings ${CRCUserName} and ${CRCPassword} are dynamically replaced with the teller logon user name and password.

The Banking Application is now configured for integration with the Siebel Financial Services 7.x Application Server.