Siebel Retail Finance Installation Guide

Version 2007

April 2007

PRODUCT MODULES AND OPTIONS. This guide contains descriptions of modules that are optional and for which you may not have purchased a license. Siebel's Sample Database also includes data related to these optional modules. As a result, your software implementation may differ from descriptions in this guide. To find out more about the modules your organization has purchased, see your corporate purchasing agent or your Siebel sales representative.

Contents

1 What’s New in This Release

What’s New in Siebel Retail Finance Installation Guide, Version 2007

Table 1 lists changes in this version of the documentation to support version 2007 of the software.

Table 1. What’s New in Siebel Retail Finance Installation Guide, Version 2007

Topic	Description
Configuring the Object Request Broker Service	This topic has been updated. The Java Object Request Broker (ORB) Service must be configured to pass parameters by reference, to optimize performance.
Deploying the Banking Application on WebSphere	This topic has been updated to reflect platform changes and changes required for Web Services deployment.
Tuning Branch Teller for Performance	This topic has been added for tuning Branch Teller.
Tuning Internet Banking for Performance	This topic has been added for tuning Internet Banking.
Enabling Support for Peripheral Devices	This topic has been updated. This topic covers enabling support for peripheral devices connected to RS232 serial ports and IEEE 1284 parallel ports.
Installing the Siebel Retail Finance Design Pack	This topic has been updated for changes required to the installation process.

2 Installing the Banking Application on WebSphere

This chapter describes how to install Oracle's Siebel Retail Finance Banking Application on the WebSphere platform. It contains the following topics:

NOTE: Refer to the *Siebel Retail Finance System Requirements and Supported Platforms* document on Siebel SupportWeb (http://supportweb.siebel.com) for information regarding the supported environments, including the supported database and application server versions.

Requirements for Installing the Banking Application on Windows

This topic 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 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 Branch Teller interfacing with Siebel Financial Services is required an installation of Siebel Financial Services is a prerequisite.

Requirements for Installing the Banking Application on UNIX

This topic 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 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 non-clustered environment, apart from the steps in this topic. 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 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 20, 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 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 commands in this document should be replaced by 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 topic details the SRF-specific procedures required when configuring WebSphere for an Oracle or DB2 database server and how to create the required database tables. Some of the topics are specific to Oracle or DB2; ignore the topics 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 topic. 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 topic, 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 topic. 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 WebSphere application server.

To start WebSphere on Windows

1. Navigate to Start > Programs > IBM WebSphere > Application Server > 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

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 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 is discussed in Configuring the JDBC Driver Path.
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 the Object Request Broker Service

The Java Object Request Broker (ORB) Service must be configured to pass parameters by reference rather than by value, to optimize performance.

To configure the Object Request Broker Service

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.OracleSequen
   ceGeneratorFactoryImpl

   mca.services.sequences.factoryClass=com.bankframe.services.sequences.DB2SequenceGe
   neratorFactoryImpl

   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

This topic assumes that the procedures for configuring the WebSphere 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

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 Select Show me all installation options *and* parameters and click Next.

The Application Deployment Options screen is then displayed.

4 Select Use default virtual host name for Web and SIP modules and click Next.

The Select Installation Options screen is displayed.

5 Select the Deploy Web Services check box.

6 Select Step 3: Provide options to perform the EJB Deploy.

7 Select Oracle V10G on the Deploy EJB option – Database type drop-down menu.

8 Click 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.

NOTE: The Application Resource warnings after step seven can be ignored.

9 Click Finish and Save to deploy the EAR file.

10 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.

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

12 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.

Enabling Support for Peripheral Devices

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

Because the Java Communications API relies on a native library, you must install it into your JRE on every client machine so that the JRE 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.

NOTE: If you re-build the Branch Teller EAR file to include the comm.jar file you do not need to install the comm API on every client machine. Refer to the *Siebel Retail Finance Banking Application Build Process Guide* for information.

To install the Java Communications API extension

1. Download the Java Communications API 2.0 from http://javashoplm.sun.com/ECom/docs/Welcome.jsp?StoreId=22&PartDetailId=7235-javacomm-2.0-spec-oth-JSpec&SiteId=JSC&TransactionId=noreg
2. Copy the win32com.dll file to the *<JRE_root>\bin* directory.
3. Copy the comm.jar file to the *<JRE_root>*\lib\ext directory.
4. Copy the javax.comm.properties file to the *jre*\lib directory.

Enabling CTR 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 and rebuild the *BranchClient* and *BranchAdministrator* projects.

To enable PDF form generation

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.

Logging in to Your Siebel Retail Finance Application

Before you start the Banking Application make sure that the Application Server and the application (siebel.ear) have started successfully.

Logging in to Branch Teller

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).

Table 2 lists the Administrator, Teller and Supervisor logons that are provided in the sample data.

Table 2. 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

Logging in to Internet Banking

Use the following URL to access the Online Registration and Internet Banking functionality:

- http://servername:port/internetbanking/

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.

Table 3 lists administrator accounts that are set up in the sample data.

Table 3. 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

Table 4 lists unregistered accounts that have been supplied in the sample data to test online registration functionality.

Table 4. 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

Logging In to the MCA Services Administration Application

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).

Tuning Branch Teller for Performance

This topic covers tuning Oracle database settings for optimum Branch Teller performance.

Tuning Database Settings for Performance

The following script needs to be executed on the Teller database for optimum Branch Teller performance:

```
execute DBMS_STATS.gather_table_stats(ownname=>'BANK', tabname => '&tablename',
estimate_percent=>dbms_stats.auto_sample_size, cascade=>TRUE, method_opt=>'FOR ALL
COLUMNS SIZE 254', degree => DBMS_STATS.DEFAULT_DEGREE);
```

Tuning Internet Banking for Performance

This section contains the following topics for tuning Internet Banking for performance:

- Configuring JDBC Connection Pools
- Configuring Thread Pool Size
- Configuring Session Management
- Configuring ADF ApplicationModule Pool Size

Configuring JDBC Connection Pools

The JDBC connection pool size should be scaled for the number of concurrent users. The database server must also be configured to fit the capacity of the Application Server. The connection pool size on the database server must be bigger than the Application Server datasource pool size.

To configure JDBC Connection Pools

1. Launch the WebSphere administration console.
2. Navigate to Resources > JDBC > Data Sources V4 > Connection Pool properties.
3. Configure the Maximum pool size setting.

Configuring Thread Pool Size

The volume of threads available on the application server has a significant impact on the performance of the application. Consult your WebSphere documentation for configuring thread pool size. The following thread pool size parameters should be configured to match your system load:

- Default
- ORB.thread.pool

- SIBFAPInboundThreadPool
- SIBFAPThreadPool
- TCPChannel.DCS
- WebContainer
- server.startup

Configuring Session Management

The Web Container session cache size should be scaled for the average number of live sessions to optimize performance. Consult your WebSphere documentation for further information.

To configure Session Management

1. Navigate to Servers > Application Servers > Server > Web Container Settings > Web Container > Session Management.
2. Configure the Session Cache Size setting.

Configuring ADF ApplicationModule Pool Size

ADF ApplicationModule (AM) pools sizes should be configured in BC4J.properties for optimum performance. Refer to ADF documentation on configuring BC4J. The following properties should be configured:

- jbo.ampool.doampooling
- jbo.ampool.initpoolsize
- jbo.ampool.maxpoolsize
- jbo.ampool.minavailablesize
- jbo.ampool.maxavailablesize
- jbo.recyclethreshold
- jbo.ampool.timetolive

3 Installing the Branch Offline Server

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

- Branch Offline Server Installation Prerequisites
- Extracting the Branch Offline Server Installation Files**Error! Reference source not found.**
- Configuring the Branch Offline Server
- Starting the Branch Offline Server
- Running Branch Teller in Offline Mode

Branch Offline Server Installation Prerequisites

Before you install the Branch Offline Server make sure that:

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

NOTE: Refer to the *Siebel Retail Finance System Requirements and Supported Platforms* document on Siebel SupportWeb (http://supportweb.siebel.com) for information regarding third party software prerequisites.

Extracting the Branch Offline Server Installation 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/BankFrameMCA/HttpServer (WebLogic)

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

   channel.http.client.url=http://<CentralServerIPAddress>:<CentralServerPortNumber>/
   BankFrameMCA/HttpServer
   ```

3. Save the changes to BankFrameResource.properties.

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.
2. Save `run.bat`.
3. Change directory to the location of your run.bat, typically `d:\BranchServer`.
4. Enter the following command:

   ```
   Run system oracle0
   ```

   where *system* and *oracle0* are the username and password for the Oracle Lite 10g database. The Offline server database and Offline Branch Server are started.

Running Branch Teller in Offline Mode

To run Branch Teller in offline mode you must start the components of Branch Teller in the following 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.

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`

4 Integrating with Siebel Financial Services

This chapter describes the integration of Siebel Retail Finance Branch Teller with Siebel Financial Services to enable Branch Teller Customer Relationship Console (CRC) functionality. It contains the following topics:

- Integration Prerequisites
- Configuring Siebel Financial Services to Enable Branch Teller CRC
- Configuring Siebel Retail Finance to Enable Branch Teller CRC

Integration Prerequisites

To enable Siebel Retail Finance Branch Teller Customer Relationship Console (CRC) functionality an installation of Siebel Financial Services is required.

Sample software configuration resources are provided in the SiebelTCRCIntegrationConfigurationResources.jar file on the Banking Application CD for each platform supported in this release. These sample integration resources should be referenced before integrating Siebel Retail Finance with Siebel Financial Services.

NOTE: Refer to the *Siebel Retail Finance System Requirements and Supported Platforms* document on Siebel SupportWeb (http://supportweb.siebel.com) for information regarding third party software prerequisites.

Configuring Siebel Financial Services to Enable Branch Teller CRC

The following Enterprise Application Integration books on the Siebel Bookshelf need to be consulted to configure Siebel Financial Services and enable the CRC components.

- *Overview: Siebel Enterprise Application Integration*
- *Integration Platform Technologies: Siebel Enterprise Application Integration*
- *Transports and Interfaces: Siebel Enterprise Application Integration*
- *Business Processes and Rules: Siebel Enterprise Application Integration*
- XML Reference: Siebel Enterprise Application Integration

Mapping Branch Teller Customers in the Siebel Database

To enable the display of a signature and photo ID for each Branch Teller customer, the customer needs to be mapped in the Siebel database. The PERSON.OWNER_ID field in the Teller database needs to be mapped to the S_CONTACT.INTEGRATION_ID field in the Siebel database.

The following SQL maps the values for all customers in the sample data:

```
UPDATE S_CONTACT set INTEGRATION_ID='100000000000222' where ROW_ID='1-AJ3J';

UPDATE S_CONTACT set INTEGRATION_ID='100000000000223' where ROW_ID='1-4WRX';

UPDATE S_CONTACT set INTEGRATION_ID='100000000000224' where ROW_ID='1-5XX0';

UPDATE S_CONTACT set INTEGRATION_ID='100000000000225' where ROW_ID='1-5T3X';

UPDATE S_CONTACT set INTEGRATION_ID='100000000000227' where ROW_ID='1-EBXS';

UPDATE S_CONTACT set INTEGRATION_ID='100000000000228' where ROW_ID='1-EH88';

UPDATE S_CONTACT set INTEGRATION_ID='100000000000229' where ROW_ID='1-E2HL';

UPDATE S_CONTACT set INTEGRATION_ID='100000000000230' where ROW_ID='1-FJNL';

UPDATE S_CONTACT set INTEGRATION_ID='100000000000231' where ROW_ID='1-7L85';

UPDATE S_CONTACT set INTEGRATION_ID='100000000000232' where ROW_ID='1-8XUJ';

Commit;
```

Configuring Siebel Retail Finance to Enable Branch Teller CRC

For full integration with Siebel Financial Services, you must configure the Banking Application with the location of the machine on which the Siebel Financial Services 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 5.

Table 5. 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.

DESTINATION_ID Column	CRC Process Description
CRC003	Get the Referrals records from the Siebel application.
CRC004	Create a Referral record in the Siebel 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 6 with the IP Address or machine name (complete with domain suffixes) of the machine hosting Oracle's Siebel Financial Services 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=TCRCQueryContact&Us
erName=${CRCUserName}&Password=${CRCPassword}', 'ON_LINE', 'False');
```

Table 6. 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

DESTINATION_ID	**CONNECTOR_FACTORY _CLASSNAME**	**CONNECTOR_ PROPERTIES**	**HOST_ STATUS**	**IS_EAB**
CRC005	com.bankframe.ei.txnhandler.connector.http.HTTPConnectionFactory	URL_STRING- http://{SIEBEL_HOST}/eai_enu/start.swe?SWEExtCmd=Execute&SWEExtSource=TCRCInsertResponse&UserName=${CRCUserName}&Password=${CRCPassword}	ON_LINE	False

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

5 Installing the Siebel Retail Finance Design Pack

This chapter describes how to install the Siebel Retail Finance design pack. This chapter contains the following topics:

- About the Design Pack
- Installing the Siebel Retail Finance Design Pack

About the Design Pack

The design pack contains all the analysis documents and design models for Branch Teller and Internet Banking. The analysis and design collateral are built for extension using Rational Rose and the Siebel Retail Finance Design Tools. For more information on the Design Tools, see the *Siebel Retail Finance Design Tools Guide.*

Installing the Siebel Retail Finance Design Pack

The following topics describe the contents of the design pack and how to install the design pack.

Contents of the Siebel Retail Finance Design Pack

The Siebel Retail Finance design pack is contained on the Siebel Retail Finance Extension Pack installation media**.** When extracted, the design Pack contains the following elements for the Branch Teller and Internet Banking modules and Entitlements sub-module:

- **A Rational Rose design model**. The model and workspace (.wsp) files are contained in the `\Rational Rose Model` folder and subfolders.
- **HTML design documents generated from the design model**. The design documents are contained in the `\RoseGeneratedDocs` folder and subfolders. The folder names are appended with Rational Rose unique identifiers, which allows for objects within groups in the design model having the same name as objects in other groups.
- **Analysis documents.** These documents are contained in the `\FrontEnd` and `\RequirementsDefinition` folders and their subfolders, and include requirement definitions, GUI documents, and use case documents.
- **An XML file exported from the generic design model.** This file, SiebelRetailFinance<*vx.x*>.xml, is contained in the `\Model XML export` folder. It is used for importing Siebel Retail Finance processes into the Screen Orchestrator and FPI tools that are part of the Siebel Retail Finance Financial Transactions Workbench.

- **Templates for consistent analysis document layout.** These templates are contained in the `\Templates` folder. The templates are HTML (.dwt) files that you can use in an HTML editor for maintaining use case and GUI documents.

NOTE: In this document, vx.x is used to refer to the current Siebel Retail Finance release.

TIP: You can navigate through all the analysis and generated design documents for a component in the release by following the hyperlinks from the Requirements Definition document for that component. These component documents are contained in the folder \RequirementsDefinition.

TIP: To browse through the generated design documents for the design model as a whole, open the `\RoseGeneratedDocs\index.html file,` which contains hyperlinks to further index pages for generated documents for the different class types (financial objects, sessions, parameter objects, and so on) in the design model.

Other Model Files Included in the Release

The design model can contain some model files for entities, sessions, and other model classes that are not loaded by the Branch Teller, Internet Banking or Entitlements workspaces, but by the workspaces for other Siebel Retail Finance modules. These model files are provided in every release to Siebel Retail Finance-licensed customers for reference purposes. The classes can be imported, if required, into an extended design model on the customer site.

Installing as the Baseline Generic Siebel Retail Finance Design Pack

This topic describes how to install the current design pack release as the initial baseline design pack for a customer project, or to completely replace a previous release with the current release, if no extension work has taken place on the customer project.

NOTE: Refer to the *Siebel Retail Finance System Requirements and Supported Platforms* document on Siebel SupportWeb (http://supportweb.siebel.com) for information regarding third party software prerequisites.

To install the analysis and generated design documents

1. Delete the analysis and generated design documents from any previous generic Siebel Retail Finance release that already exist in the workspace.
2. Extract the design pack zip file to a temporary folder.
3. Extract the contents of the `\FrontEnd`, `\RequirementsDefinition`, `\RoseGeneratedDocs` and `\Templates` folders in the release zip file to the project workspace as a whole unit. Make sure that you retain the relative position of analysis and generated design documents using the folder structures in the release .zip file, so that hyperlinks between analysis documents and generated documents continue to work.

To install the design model

1. Delete the model files from any previous generic Siebel Retail Finance release that already exist in the customer's workspace.

   NOTE: The files from a previous release can be labeled and stored in the customer's configuration management system for the project.

2. Extract the design pack .zip file to a temporary folder.
3. Extract the `\Rational Rose Model\Siebel Retail Finance Model` folder and all its contents (including subfolders) from the release .zip file to the designated root folder for the design model in the project workspace. The root folder can have any name and path, such as `d:\vault` or `c:\model`. You must retain the relative structure of the subfolders that is present in the release .zip file.

   NOTE: The .mdl (model) and .wsp (workspace) files load the model .cat files in the release from a subfolder structure that is relative to the model and workspace file, therefore folder structures must be maintained relative to these files.

Opening the Siebel Retail Finance Generic Design Models

You use the following workspace files together to open the Branch Teller model:

- teller.wsp
- entitlements.wsp

You use the following workspace files together to open the Internet Banking model:

- e-banking.wsp
- entitlements.wsp

To open the Siebel Retail Finance generic design model

1. Navigate to the `\Rational Rose Model\Siebel Retail Finance Model` folder.
2. Double-click the teller.wsp or e-banking.wsp file.
3. Double-click the entitlements .wsp file.

   The Entitlements sub-module workspace is loaded and the Entitlements classes are added into the Branch Teller or Internet Banking model.

CAUTION: You must open the banking application model by opening the workspace (.wsp files) files and not by opening the Rational Rose model file (the .mdl file), which loads many model classes that are not required by individual banking application projects.

6 Installing the Financial Components WorkBench

This chapter covers installation of the tools which comprise the Siebel Retail Finance Financial Components WorkBench. It contains the following topics:

- Installing the Design Tools
- Installing the Screen Orchestrator
- Installing the Financial Process Integrator

Installing the Design Tools

This topic describes how to install the Siebel Retail Finance Design Tools. It contains the following tasks:

- Design Tools Installation Prerequisites
- Migrating from a Previous Version of the Design Tools
- Installing the Design Tools
- Design Tools Run-Time Requirements
- Setting the Screen Resolution

Design Tools Installation Prerequisites

Before installing the Design Tools the following must apply:

- You must have administrator privileges on the Design Tools installation machine.
- Rational Rose must be installed on the Design Tools installation machine.
- All applications, including Rational Rose, must be closed as registry settings are updated during the Design Tools installation process.

NOTE: Refer to the *Siebel Retail Finance System Requirements and Supported Platforms* document on Siebel SupportWeb (http://supportweb.siebel.com) for information regarding third party software prerequisites.

Migrating from a Previous Version of the Design Tools

If you have a previous version of the Design Tools installed, you must uninstall that version, install the new version, and update the attributes of all existing models with the unique ID for any associated common validators. This unique ID is used to find the grouping for the validator when the XML representation of the model is generated.

To uninstall a previous version of the design tools

1. Navigate to the Windows Control Panel.
2. Select Add or Remove Programs.
3. Select Siebel Design Tools.
4. Click Change/Remove.

Installing the Design Tools

This topic describes the Design Tools installation process.

To install the Design Tools

1. Navigate to the installation setup file, which is named Siebel Design Tools (Delivery) *version number*.exe, and double-click on this file.

   NOTE: If you have received the software on CD, a link to the installation setup file might be displayed automatically; click on the link to start the Installation Wizard.

2. Follow the instructions in the Installation Wizard.
3. Restart, if you are prompted to do so.
4. After installation, if the Design Tools do not appear under Siebel on the Tools menu in Rational Rose, or if selecting this menu option generates an error, select Siebel Design Tools in the Rational Rose Add-In Manager drop-down list.

After you install the Design Tools, you must update any existing models to assign unique identifiers to their attributes.

To update the attributes of existing models

1. Start Rational Rose with the model loaded.
2. Navigate to the Design Tools directory, typically `D:\Program Files\Rational\Rose\Siebel Design Tools`.
3. Run updateattributes.exe.
4. Repeat steps 1 through 3 for all existing models.

Design Tools Run-Time Requirements

Before you run the Design Tools, you must install and configure the third-party software, Microsoft XML Parser, which the Model Comparison Tool uses to compare the XML representations of two models.

To install Microsoft XML parser

1. Download the MSXML 3.0 Service Pack 4 (msxml3usa.msi) from http://download.microsoft.com/download/d/9/8/d9886528-6438-4828-9094-697103203a32/msxml3usa.msi.

2. Double-click the downloaded msxml3usa.msi file, and follow the instructions in the Installation Wizard.

Setting the Screen Resolution

For optimum display, set the screen resolution of the machine on which you install the Design Tools to 1024 x 768 pixels.

Installing the Screen Orchestrator

This topic covers installation of the Siebel Retail Finance Screen Orchestrator.

Screen Orchestrator Installation Prerequisites

- Before you install Screen Orchestrator, you must install and configure the Java Runtime Environment (JRE).

- The Adobe Acrobat Viewer must be installed to view the Screen Orchestrator Help file.

NOTE: Refer to the *Siebel Retail Finance System Requirements and Supported Platforms* document on Siebel SupportWeb (http://supportweb.siebel.com) for information regarding third party software prerequisites.

Migrating From a Previous Version of Screen Orchestrator

If you have a previous version of the Screen Orchestrator installed, you must uninstall that version and save any work before you install the new version.

To migrate from a previous version of Screen Orchestrator

1. Save to an archive location any work in subdirectories of the directory where Screen Orchestrator is installed.

2. Navigate to the setup.exe file, in the root directory on the CD, and double-click on the setup.exe file.

   Screen Orchestrator is uninstalled.

NOTE: The processes.xml file is not deleted from the resources directory. This file must be preserved when migrating, because the contents of the file change as new processes are added using the Screen Orchestrator.

Installing the Screen Orchestrator

To install the Screen Orchestrator, double-click the setup.exe file and follow the instructions in the Installation Wizard.

Screen Orchestrator Run-Time Requirement

After you install Screen Orchestrator, make sure that the JAVA_HOME variable in the sethomes.bat file points to the location of your JRE. The sethomes.bat file is located in the root of the directory where you install the Screen Orchestrator.

Setting the Screen Resolution

For optimum display, set the screen resolution of the machine on which you install Screen Orchestrator to 1024 x 768 pixels.

Installing the Financial Process Integrator

This topic covers installation of the Siebel Retail Finance Financial Process Integrator.

Financial Process Integrator Installation Prerequisites

Before installing the Siebel Financial Process Integrator the following must be installed or configured:

- Sun Microsystems JRE must be installed before installing the Siebel Financial Process Integrator.
- The system variable JAVA_HOME must be configured to point to the JRE.
- The Adobe Acrobat Viewer must be installed to view the Financial Process Integrator Help file.

NOTE: Refer to the *Siebel Retail Finance System Requirements and Supported Platforms* document on Siebel SupportWeb (http://supportweb.siebel.com) for information regarding third party software prerequisites.

Migrating from a Previous Version of the Financial Process Integrator

To migrate from a previous version of the Financial Process Integrator

1. Save any modified FPI Workspace files to an archive location.
2. Double-click on the Financial Process Integrator installation exe and follow the installation wizard instructions to uninstall the previous version. The FPI is then uninstalled.

Installing the Financial Process Integrator

- Click on the Install Financial Process Integrator link on the Financial Transactions WorkBench installation CD and follow the on-screen instructions.

Financial Process Integrator Runtime Requirements

This topic covers configuring the FPI classpath and setting the screen resolution.

Configuring the FPI Classpath for Database Drivers

The database driver needs to be added to the Financial Process Integrator's classpath to execute SQL on a target database at run time. Configuring the classpath for database drivers or connecting to a database is not required to generate SQL from the FPI.

To add the database driver to the Financial Process Integrator's classpath

1. Select Start -> Programs -> Siebel.
2. Right click on the Siebel Financial Process Integrator menu item and select Edit. The FPI run.bat file is then launched in a text editor.
3. Add the entry for the database driver to run.bat after the –classpath variable. The typical classpath entries for DB2 and Oracle are listed in the table below.

Database	Driver Classpath Entry
DB2	D:\ Program Files\IBM\SQLLIB\java\db2java.zip;
Oracle	D:\oracle\ora92\jdbc\lib\classes12.jar;

Setting the Screen Resolution

The screen resolution of the machine on which Oracle's Siebel Retail Finance Financial Process Integrator has been installed should be set to 1024 x 768 pixels for optimum display.