Oracle® Applications

Supplemental CRM Installation Steps

Release 11*i*

October 2000

Part No. A86291-01

ORACLE®

Supplemental CRM Installation Steps, Release 11*i*

Part No. A86291-01

Contents

Send Us Your Comments

Supplemental CRM Installation Steps, Release 11*i*

Part No. A86291-01

Oracle Corporation welcomes your comments and suggestions on the quality and usefulness of this document. Your input is an important part of the information used for revision.

- Did you find any errors?
- Is the information clearly presented?
- Do you need more information? If so, where?
- Are the examples correct? Do you need more examples?
- What features did you like most?

If you find any errors or have any other suggestions for improvement, please indicate the document title and part number, and the chapter, section, and page number (if available). You can send comments to us in the following ways:

- Electronic mail: chad.mack@oracle.com
- FAX: telephone number (650) 654-6223 Attn: Oracle CRM Operations
- Postal service:
  Oracle Corporation
  Oracle CRM Operations
  500 Oracle Parkway, 6op8
  Redwood Shores, CA 94065
  USA

If you would like a reply, please give your name, address, telephone number, and (optionally) electronic mail address.

If you have problems with the software, please contact your local Oracle Support Services.

Preface

Supplemental CRM Installation Steps provides instructions for completing your installation of Oracle Customer Relationship Management (CRM) products. In this release of Oracle CRM, much of the installation process is handled using the new Oracle Rapid Install product, which automates many of the required steps and minimizes the time it takes to implement Oracle CRM Applications and the Oracle8i Server technology stack.

If you are responsible for implementing Oracle CRM Applications, it is important that you read and understand the information in *Oracle Applications Concepts* as well as the information in this guide. *Oracle Applications Concepts* explains the technology, architecture, and terminology used in this document.

Intended Audience

This document is contains tasks to be performed by the following IT professionals:

Database Administrator

Installs and configures the Oracle database and maintains database access controls. This person provides consultation on performance and is responsible for monitoring growth and fragmentation of the production database and ensuring database backup and recovery.

System Administrator

Responsible for administering the development system. This person's responsibilities include:

- Ensuring that hardware is correctly configured
- Installing, configuring, and maintaining operating and development software

- Ensuring that the system is backed up daily
- Designing and maintaining system security — for example, establishing system accounts.

The system administrator provides first-line support for problems with the development system and ensures that faults are quickly rectified. They may perform the setup and initial maintenance of the production system or advise the client's operational staff on these tasks. The system administrator works with the project team to optimize system performance, install packaged applications environments, and convert data.

Technical Specialist

Responsible for designing, developing, unit testing, implementing, and maintaining the custom extensions for Oracle CRM Applications. These extensions include, but are not limited to, modules such as interfaces, automated data conversions, reports, forms, and enhancements.

Related Documents

The following manuals on the Oracle Applications, Release 11*i* Documentation CD-ROM are referenced in this document:

- *Oracle Applications Concepts*
- *Installing Oracle Applications, Release 11i*
- *Oracle Call Center Applications Setup*
- *Oracle CRL Financials Implementation Guide*
- *Oracle CRL-Financials Enabled Projects Concepts and Procedures*
- *Oracle Mobile Device Gateway Concepts and Procedures*

CRM Applications Supplementary Installation Tasks

This chapter provides instructions for completing your installation of Oracle CRM applications. Perform the supplementary steps described here **after** the general installation procedures described in *Installing Oracle Applications, Release 11i.*

> **Note:** The Oracle CRM applications not listed in this manual do not require additional installation tasks.

This manual details the supplementary installation tasks for the following Oracle CRM applications:

Product Family	Application(s)
CRM Tools and Technology	■ HTML Stack ■ Common MES
Internet Business Applications	■ Oracle iSupport ■ Oracle iPayment ■ Oracle eMail Center
Marketing Applications	■ Oracle Marketing Encyclopedia System
Call Center Applications	■ Oracle Telephony Manager ■ Oracle Scripting
Communications and Utilities Applications	■ Oracle Network Logistics - NATS ■ Oracle SDP Provisioning ■ Oracle CRL Financials

CRM Tools and Technology

This section includes the supplemental installation tasks for the following Oracle CRM Foundation components:

- HTML Stack
- Common MES

HTML Stack

Supplemental Installation Steps

1. In order for Oracle CRM Foundation to identify the name and location of the dbc file and location of the log files, three parameters need to be sent to the Java Virtual Machine as "-D" parameters:

   - -DJTFDBCFILE=<name of dbc file>
   - -Dframework.Logging.system.filename=<name of framework log file>
   - -Dservice.Logging.common.filename=<name of application log file>

   When running Oracle Applications using Apache server and Jserv processes, send the above three parameters to each of the Jserv processes. Accomplish this by adding these "-D" parameters to the "java" command in the jservctl file.

   Below is an example of what the file jservctl needs to contain:

```
#define variable for -DJTFDBCFILE
ConfFile="-DJTFDBCFILE=/u03/tools/apache/bal_config/foo.dbc"

#define variable for -Dframework.Logging.system.filenam
FWLogFile="-Dframework.Logging.system.filename=/u03/tools/apache/bal_
config/logs/fw_log"

#define variable for -Dservice.Logging.common.filename=
JTFLogFile="-Dservice.Logging.common.filename=/u03/tools/apache/bal_
config/logs/jtf_log"

#pass the above three variables to the jvm, along with any other arguments
$JDK_HOME/bin/java -mx221m  $ConfFile $FWLogFile $JTFLogFile  -classpath
$classes org.apache.jserv.JServ $props >> $log  2>&1   &
```

2. If your apache server is running on a UNIX platform, then set the file handle limit before restarting the apache server . Depending on the UNIX shell you are running, the command will be a variation of the *ulimit* **or** *limit* command*:*

   a. At the command line, verify the limit for descriptors (e.g., *ulimit -n* **or** *limit*).

   b. Set the limit for descriptors to 1024 (e.g., *ulimit -S -n 1024* **or** *limit descriptors 1024*).

   c. Verify your changes.

Common MES

Using Oracle CRM Foundation requires recreating index on URL_STRING column in JTF_AMV_ITEMS_B table to add the proxy settings information specific to the user site. Run jtfiimt.sql using Sql*Plus for each JTF schema, passing the following parameters.

- JTF username
- JTF password
- site proxy server
- comma-delimited list of domain names where proxy does not have to be used

Use the following syntax:

For Unix Users:

```
$ cd $ JTF_TOP/patch/115/sql/
$ sqlplus <username>/<JTF password> @jtfiimt.sql <JTF username> <JTF password>
<proxy server> '<list of sites not requiring proxy>'
```

For NT Users:

```
C:\> cd %JTF_TOP%\patch\115\sql
C:\> sqlplus <JTF username>/<JTF password> @jtfiimt.sql <JTF username> <JTF
password> <proxy server> '<list of sites not requiring proxy>'
```

Example:

```
sqlplus JTF/JTF @jtfiimt.sql JTF JTF 'proxy-server.foobaa.com' 'foobar.com,
us.foobar.com'
```

Once the above script is successfully run, the proxy setting preferences are created, the user now needs to call the script jtfiaibu.sql using Sql*Plus for each JTF schema, this script drops and recreates index on URL_STRING column in JTF_AMV_ ITEMS_B table, passing the following parameters:

- JTF username
- JTF password

Use the following syntax:

For Unix Users:

```
$ cd $ JTF_TOP/patch/115/sql/
$ sqlplus <username>/<JTF password> @jtfiaibu.sql <JTF username> <JTF password>
```

For NT Users:

```
C:\> cd %JTF_TOP%\patch\115\sql
```

3. `C:\> sqlplus <JTF username>/<JTF password> @jtfiaibu.sql <JTF username> <JTF password>`

Internet Business Applications

This section includes the supplemental installation tasks for the following Oracle CRM applications:

- Oracle iSupport
- Oracle iPayment
- Oracle eMail Center

Oracle iSupport

Supplemental Installation Steps

To enable survey/feedback, the merchant must set up the survey configuration in iSupport System Administration. The merchant must construct the survey/feedback through the Scripting Author in Oracle Scripting. Please refer to *Oracle Scripting Concepts and Procedures* for details.

The iSupport product uses Oracle Scripting to run the survey, so the merchant must set up the system configuration for the Scripting module.

The system configuration includes:

- The server name on which the Scripting engine resides
- The port number
- The database name on which the scripts are stored
- Java Native Directory Interface
- Username
- Password

Refer to *Oracle Call Center Applications Setup* on the Oracle Call Center Applications Setup CD-ROM for Oracle Scripting installation documentation.

Oracle iPayment

Installing and Configuring iPayment Java Servlets

Oracle iPayment has several Java servlets which are not configured as part of Oracle Applications Rapid Install process. Follow the step-by-step instructions in this section to configure these servlets.

The instructions in this section assume you have some knowledge of configuring Java servlets with Apache Web Server. In particular, the instructions assume you know where to find Apache and JServ configuration files on the node where the Apache Web Server is installed. For further information, refer to the Apache documentation available at www.apache.org.

1. Login to the Web Server node.

   Log on to your Web Server node as the applmgr user and run the environment file to set up the Oracle Applications environment. Your environment should have the following variables defined:

   ```
   $AU_TOP, $FND_TOP, $IBY_TOP
   ```

   $AU_TOP is the Applications utilities top-level directory, which contains common applications files.

   $FND_TOP refers to the top-level directory where foundation services are located (foundation service is a group of basic facilities common across many products).

   $IBY_TOP refers to the top-level directory of Oracle iPayment installation.

Note: Apache and Jserv may not interpret environment variables in their configuration files; please expand any variables used below of the form $ABC to the values they actually contain on your installation.

For example, if $IBY_TOP is defined at /u03/apps/iby/11.5, you need to replace $IBY_TOP with "/u03/apps/iby/11.5" in the instructions below.

2. Verify that you already have a common Servlet Zone configured in your environment.

   A servlet zone should already exist in your Apache Web Server installation. Check jserv.properties for a line beginning with "zone=". If you see such a line, a servlet zone has been set up. This zone is referred to as <SERVLET_ZONE> in this document. You must replace <SERVLET_ZONE> with your actual zone name.

3. Configure the ECApp Servlet.

   An ECApp servlet is needed in order to use the PL/SQL API of iPayment and for iPayment 3i backward-compatibility API.

   Set up virtual path mapping for ECApp Servlet.

   Add the following line to your zone property file, <SERVLET_ZONE>.properties (remember to replace <SERVLET_ZONE> with actual zone name).

   ```
   servlet.ecapp.code=oracle.apps.iby.ecservlet.ECServlet
   ```

   This allows the ECAppservlet to be invoked as http://<hostname>:<port>/iby/ecapp where:

   <hostname> is the name of the server that you are running iPayment, and <port> is the port number for ECAppservlet.

4. Configure the CyberCash Payment System Servlet.

   CyberCash Payment System Servlet is only needed if you are planning to process credit card and bank transfer payments through the CyberCash Service. For more information refer to the section on Payment Systems in *Oracle iPayment Concepts and Procedures* or in the Understanding iPayment section of the application online help.

Please follow the steps below to configure Cybercash Merchant Connection Kit, also known as MCK to work with Oracle iPayment.

a. Set up a merchant account with CyberCash at http://amps.cybercash.com if you do not already have one.

b. Download CyberCash's Merchant Connection Kit (MCK) from http://cr.cybercash.com. Follow CyberCash's instructions to install the MCK.

c. Go to the directory where the MCK C libraries are located. The installation directory should be named mck-<version>-<operating system>. For example, if you installed MCK version 3.2.0.6 on Solaris under the /usr/oracle directory, you would navigate to the following:

```
% cd /usr/oracle/mck-3.2.0.6-sparc-sun-solaris2.6/c-api/lib
```

d. Copy the three MCK libraries mentioned below into the $IBY_TOP/lib directory:

```
% cp libCCMck.a $IBY_TOP/lib
% cp libmckcrypto.a $IBY_TOP/lib
% cp libmd5hash.a $IBY_TOP/lib
```

e. Uncomment the following lines from $IBY_TOP/admin/driver/ibysub01.drv:

```
#  iby          bin          libcybnv.so
#  iby          bin          libcybnv.dll
```

so that the lines read:

```
   iby          bin          libcybnv.so
   iby          bin          libcybnv.dll
```

f. Build the Interactive adapter executable by running adadmin.

Select the option to "Relink Applications Programs", and enter values as follows:

Prompt	Value
Enter list of products to link ('all' for all products) [all]:	iby
Generate specific executables for each selected product [No]?	no

Successful completion of the above step will build the required executable in $XDP_TOP/bin. In case of errors, contact your Oracle Support representative.

g. Set the wrapper.env variable in the file jserv.conf as follows:

```
wrapper.env=LD_LIBRARY_PATH=$IBY_TOP/lib
```

If there is already a line wrapper.env=LD_LIBRARY_PATH=..., then append the above location as you would with the LD_LIBRARY_PATH environment variable.

For example, if you have a line

```
wrapper.env=LD_LIBRARY_PATH=$ABC/lib
```

add `:$IBY_TOP/lib` at the end of line. The result should be

```
wrapper.env=LD_LIBRARY_PATH=$ABC/lib:$IBY_TOP/lib
```

h. Set up a virtual path mapping for CyberCash servlet.

Insert the following line in the zone property file <SERVLET_ZONE>.properties which is typically located in the etc directory of your top Jserv engine directory (e.g., /d1/testcomn/util/apache/1.3.9/Apache/Jserv/etc): servlet.oramipp_cyb.code=oracle.apps.iby.bep.cybercash.CybServlet.

This allows the servlet to be invoked as:
http://<hostname>:<port>/<servlet_zone>/oramipp_cyb

i. Set the servlet init parameters.

There are several initialization parameters that are recognized by the Oracle iPayment Cybercash Servlet. Set these init parameters by inserting the following line in the zone property file <SERVLET_ZONE>.properties file:

```
servlet.oramipp_cyb.initArgs=mckhome=$MCK_HOME,debug=false,logfile=$IBY_TOP/log/ibycybserv.log
```

The following are the initialization parameters recognized by the Cybercash Servlet:

mckhome

This parameter is mandatory. It's the directory path that points to the location where the CyberCash Merchant Connection Kit is installed. For

example, if a merchant named test-mck has been installed such that its associated files can be found under the directory /usr/oracle/mck/test-mck, then mckhome should be set to /usr/oracle/mck.

Transaction requests to iPayment will fail if mckhome is not set correctly.

debug

This parameter is optional. If set to "true" then the servlet will print debugging information to the body of its responses in plain text. This information includes the inputs sent to the servlet during the request, as well as the outputs the servlet sends for its response. If an exception is thrown during the processing of the request then a stack trace is also printed.

logfile

This parameter is optional. It's a string which specifies the fully qualified path name of the log file location. The input and output values of each transaction are written to this file, as well as stack traces if an exception is thrown. If this parameter is not set, logging will be turned off.

5. Configure the Scheduler Servlet.

   This step is required if you want to set up a scheduler in iPayment. A scheduler is required if you process off-line payment operations.

   Set up virtual path mapping for Scheduler servlet by adding the following line to the zone property file <SERVLET_ZONE>.properties:

   ```
   servlet.scheduler.code=oracle.apps.iby.scheduler.PSReqHandler
   ```

   This allows the servlet to be invoked as:

   http://<hostname>:<port>/iby/scheduler

Oracle eMail Center

Supplemental Installation Steps

1. Install Oracle Email Server (OES) 5.1 (patchset 3) on the same or on a separate instance as the Oracle Applications Instance. Follow the instructions provided in the *Oracle Email Server Installation Guide* (Installing Oracle Email Server section). Oracle Email Server was formerly known as Oracle Internet Messaging (IM).

In the post-install Email Server configuration, select **NO** for LDAP install. For more information on this step, refer to the *Oracle Email Server Installation Guide* (Configuring Oracle Email Server section).

Once the installation process is complete, you can configure the Oracle Email Server instance to talk with the Oracle Applications Instance.

2. Connect to the Oracle Email Server database instance with the **OO** user and password. Grant privilege on IM_IMT_EXTN to oraoffice.

> **Note:** The default configuration is single instance. Single instance implies that Oracle Applications data and Oracle Email Server data reside in the same database.

3. Create a sqlnet tnsnames entry on the Oracle Email Server machine to point to the Oracle Applications Instance. (If installed on an instance separate from Oracle Applications Instance).
4. Create a Database Link to the CRM instance from the Oracle Email Server instance. You need the following:
   - Password for the Oracle Applications APPS schema.
   - Password for the Oracle Email Server OO schema.

   Use SQLPLUS to login to Oracle Email Server as user **OO** (using the **OO** password entered during Oracle Email Server installation) and use the following to create the database link:

   CREATE DATABASE LINK <link name> CONNECT TO <APPS username> IDENTIFIED BY <APPS password> USING <tns / service name>;

> **Note:** The name of the above mentioned database link <link name> is used later to configure other eMC components.

Email Server Installation

Install Oracle Email Server either in the same APPS database or in a separate database on a different machine. Perform steps 1-4 in both scenarios.

Scenario 1: OES Installed in the APPS Database

1. The following OES components must be installed per the instructions provided in the OES installation manual.
   - Oracle Email Server 5.1.0.0.1
   - IMAP4 and POP3 protocol servers 5.1.0.0.1
   - Email Administrator 5.1.0.0.1
2. Email Server Post-Installation Tasks
   a. Login as root and run root.sh from $ORACLE_HOME.
   b. Change the ownership of $ORACLE_HOME/bin/ofcpsl file to root.
   c. Modify the imconfig file to point to the correct JRE_HOME (JRE1.1.8) directory.
3. IM configuration
   a. Login as oracle (OS user).
   b. Set the DISPLAY environment variable according to your Unix shell.

      e.g., setenv DISPLAY <local terminal>:0.0

Shell	Example
csh	setenv DISPLAY crmops.us.oracle.com:0.0
sh, ksh	DISPLAY= crmops.us.oracle.com:0.0; export DISPLAY

   c. Run imconfig under $ORACLE_HOME/bin.
   d. Use the table below to enter the required information at the configuration prompts:

Prompt	Recommended Values or Guideline
Enter DomainName	Example: BARRACUDA_DOMAIN
Email Server Node	Choose either default node or custom node. (If uncertain, use the default email server node.)
First Oracle Email Server Node	yes
Enter Node name	Example: BARRACUDE_NODE

Prompt	Recommended Values or Guideline
Set Password for User admin Password	Example: BARRACUDA
Set Password for Database Users	■ OO password (ex: OO) ■ oraoffice password (ex: oraoffice) ■ OO_DS user password (ex: OO_DS) ■ OO_MAIL user password (ex: OO_MAIL) ■ OO_PUBLIC user password (ex: OO_PUBLIC) ■ OO_SCHED user password (ex: OO_SCHED)
Do you Want to Enable IMAP4 for Default SMTP Gateway?	yes
New Gateway Name	Use the default (smtp)
Enter Email Domain Name	Example: <db machine name>.us.oracle.com
ldap	no

4. Sendmail Configuration (UNIX side)
   a. Copy the sendmail.cf file to /etc/mail.
   b. Edit sendmail.cf to modify ORACLE_HOME, ORACLE_SID to OES database home, and database SID.
   c. Use the following commands to check the ClassID:

```
cd $ORACLE_HOME/bin (OES DB)
ofcguard start (start/status/stop)
oomgr admin/<password> (ex: BARRACUDA)
IOFCMGR> show gateway all;
```

      The ClassID displays.

   d. Check the ClassID from the database and put the same ClassID in the sendmail.cf.
5. Add user group aliases in the /etc/aliases file.

Scenario 2: OES Installed in a Separate Database

Perform the following steps when you install Oracle Email Server in a separate database.

1. A tns-service entry for the APPS database needs to be added in tnsnames.ora (in /etc or /var/opt/oracle) file in OES DB machine.
2. A tns-service entry for the OES database needs to be added in tnsnames.ora (in /etc or /var/opt/oracle) file in the APPS DB machine.
3. Verify that the SQL*Net connections from both databases work correctly.
4. Create the following database links in your Oracle Applications database:
   a. Using SQL*Plus log in as the APPS user.
   b. Create the database link <IMLINK>:

      ```
      connect to oraoffice identified by <oraoffice password> using '<tns
      service name of OES dbname >';
      ```

   c. Create a database link to the <OOLINK>:

      ```
       create database link <OOLINK> connect to <OO schemaname> identifed by
      <OO password> using  '<tns service name of OES dbname>';
      ```

5. Create the following database link in OES database:
   a. SQL*Plus login as OO, using OO password that was entered during OES installation.
   b. Create a database link <appslink> connecting to apps identified by <apps pas word> using <tns/ service name of APPS dbname>.
6. SQL*Plus login to OES DB as user OO_MAIL using OO_MAIL password that was entered during OES installation.

   ```
   grant select on om_imt_msgpart to oraoffice;
   ```

Oracle Email Center Template Hosting Post-Process Installation Steps

The Email Center (eMC) product supports the use of web forms to generate structured email messages to the Email Center server. (See the eMC documentation for more information on this feature.) To make use of web forms you must first configure the web server to use the Java servlet installed with the eMC product. In cases where the web server does not support Java servlets, the application will default to the use of a CGI script to process the web form and generate the

appropriate email message. The performance benefits of using the Java servlet make it the optimal choice.

Perform the following tasks to generate structured email messages using web forms:

1. Determine if your web server supports Java servlets.
2. Determine the location of the web server Java servlet properties file.
3. Configure the web server for the eMC email generation servlet.

The first step to configuring your web server is to determine if the web server supports Java servlets. Most web servers support Java servlets. If your web server does not support Java servlets, then you will have to use the supplied CGI script equivalent on your web server.

When using the Java servlet there is one manual step required for hosting the inbound eMC templates for structured email processing. That is the registering of the eMC Java servlet with the customer web server. The web server has to be capable of hosting servlets. Each web server handles servlets in its own way. Sun's web server has native servlet support. Apache Web Server supports servlets with the add-on Jserv module. Check with your web server administrator for details on your web server's support for servlets.

If your web server does not support servlets then you will need to install the CGI equivalent of the servlet on your web server. Both the servlet and CGI script are included with the 1 to 1 Fulfillment installation (emailgeneratingservlet and emailgeneratingCGI). No configuration steps are required when using the CGI.

After verifying servlet support on the web server, you will have to configure the web server servlet settings. The property settings for servlets are unique to each web server. For example, with Apache Web Server using the Jserv servlet engine. The configuration information needs to be updated in the zone.properties file that is usually found in the Apache Jserv conf directory. Adding the entries given below to the file will configure the servlet.

Example:

```
# Servlet Parameters
# Startup Servlets
servlets.startup=emailgen

# Servlet Aliases
servlet.emailgen.code=EmailGeneratingServlet

# Aliased Servlet Init Parameters
```

```
servlet.emailgen.initArgs=hostname=<outgoing SMTP server>,to=<receiving account
for eMC email processing>
```

The parameter values remain the same even if a different web server is used. These parameters are defined as:

- **emailgen** – Name of the servlet
- **EmailGeneratingServlet** – servlet class name
- **Outgoing SMTP server** – SMTP server in the organization that is used to send outgoing emails
- **Receiving account for eMC email processing** – inbound email account name setup for eMC email processing

Marketing Applications

This section includes the supplemental installation tasks for the following Oracle CRM application:

- Oracle Marketing Encyclopedia System

Oracle Marketing Encyclopedia System

Supplemental Installation Steps

1. Using Oracle Marketing Encyclopedia requires modifications to Oracle Workflow directory services views (WF_ROLES and WF_USER_ROLES) to include Oracle Marketing Encyclopedia roles and role users.

   To accomplish this, please apply the Oracle Workflow patch for the Bug # 1409680 and make sure that the patch is applied successfully.

2. Start the concurrent program MES Matching Engine. This program publishes items into the repository.

Call Center Applications

This section includes the supplemental installation tasks for the following Oracle CRM applications:

- Oracle Telephony Manager
- Oracle Scripting

Oracle Telephony Manager

Implementation Requirements

Oracle Telephony Manager uses the Oracle Universal Installer to perform additional configuration steps on the web server node. You will need the *Oracle Call Center Applications Setup* CD in order to perform these steps.

Modules on the *Oracle Call Center Applications Setup* CD include:

- Oracle Scripting Author
- Oracle Telephony Manager and subcomponents (Telephony Media Control, Inbound Telephony Server, Routing Server, and Server Monitor)
- Oracle Email Center
- Oracle Universal Work Queue
- Oracle Interaction Blending

Refer to *Oracle Call Center Applications Setup* on the Oracle Call Center Applications Setup CD-ROM for additional implementation documentation.

Oracle Scripting

Supplemental Installation Steps

1. Publish the JNDI name for Oracle Applications server objects:

```
publish -republish -user <APPS-username> -password <APPS-password>
-service sess_iiop://localhost:<IIOP port>:<SID>
/test/oracle/apps/ies/corba/common/Master
oracle.apps.ies.corba.server.MasterImpl
oracle.apps.ies.corba.common.MasterHelper
```

   Details are as follows:

   - "-user <APPS-username> -password <APPS-password>"specifies the schema where the JNDI name should be published
   - "-service sess_iiop://localhost:<IIOP port>:<SID>" specifies the URL of the 8i database, with a valid IIOP listener port, and SID
   - "-republish" allows this command to succeed even if the JNDI name already exists

- "/test/oracle/apps/ies/corba/common/Master" is the JNDI name to be published
- "oracle.apps.ies.corba.server.MasterImpl" is the Java class to be invoked when the JNDI name is called
- "oracle.apps.ies.corba.common.MasterHelper" is a helper Java class that is also needed

2. Load Applications JAR files into the database. Using the loadjava utility, load the following JAR files into your database. Log in as the oracle user and make sure your ORACLE_HOME refers to an 8.1.6 Oracle home and ORACLE_SID refers to your database.

   Details are as follows:

   - $ cd $JAVA_TOP/oracle/apps/ies/jar
   - $ loadjava -user APPS/APPS -resolve -oracleresolver -synonym -definer -oci8 iescommn.jar
   - $ loadjava -user APPS/APPS -resolve -oracleresolver -synonym -definer -oci8 iesservr.jar

3. Apply Server Technologies (ST) patch

   Please apply the backported patch for ST bug 1199486 for your platform. The backport for Sun Solaris is available as patch 1309233. Backports on other platforms may be available at a later date.

4. Apply EWT patch

   Apply the ARU patch 1358591. This ARU contains a readme with extensive instructions for installing EWT version 3.3.10, which must be downloaded from tcpatch. Make sure to carefully read and follow the instructions in the readme.

5. If you intend to install Oracle Scripting Author, refer to *Oracle Call Center Applications Setup* on the Oracle Call Center Applications Setup CD-ROM for additional implementation tasks.

Communications and Utilities Applications

This section includes the supplemental installation tasks for the following Oracle CRM applications:

- Oracle Network Logistics - NATS
- Oracle SDP Provisioning

- Oracle CRL Financials

Oracle Network Logistics - NATS

Using Oracle Network Logistics requires enabling triggers CUN_MASS_ADDITIONS_ARU and CUN_FA_RETIREMENTS_ARI. Accomplish this by running the cuntrg03.sql script on the admin tier.

Using SQL*Plus, run the script cuntrg03.sql against each APPS user:

```
sqlplus <APPS username>/<APPS password> @$CUN_TOP/patch/115/sql/cuntrg03.sql
```

Oracle SDP Provisioning

Oracle Provisioning's interactive adapter uses a third-party software called Expect. Expect is a tool used primarily for automating and testing interactive applications such as telnet, ftp etc. Expect is packaged as a set of utilities that can be used as interpreted scripts and/or as a library that can be linked with the 'C' objects to form an executable. Additionally, Expect is free and in the public domain.

Oracle Provisioning's interactive adapter uses Expect APIs and is required to be linked with the Expect and Tcl libraries. Expect uses Tcl, a general purpose scripting language, internally. Expect libraries are not distributed platform-wise. Instead, Expect comes packaged as a compressed TAR or ZIP file. Expect is supported on most Posix-compliant operating systems. Most Unix and other popular platforms conform to Posix standards. The Expect package must be uncompressed on the target platform, and its library must be built using the installation notes which are included in the package.

Oracle does not distribute Expect related libraries along with its products. You are required to download, build and optionally install Expect and Tcl libraries at your site. After required libraries are successfully built, interactive adapter executable should be built using the steps documented in this section.

Building the Interactive Adapter Executable

The following required steps build the interactive adapter executable, XDPNTRCT for Unix platforms:

1. Download and build Expect and Tcl libraries, libexpect5.30.a and libtcl8.0.a, in Expect and Tcl staging areas as per installation instructions enclosed in their respective packages. More details on Expect are available at http://expect.nist.gov.

2. Copy the Expect and Tcl libraries to $XDP_TOP/lib directory on the Concurrent Processing (node) Tier.
3. Uncomment the following line from $XDP_TOP/admin/driver/xdpsub01.drv:

```
# xdp          bin          XDPNTRCT
```

so that the line reads:

```
xdp          bin          XDPNTRCT
```

4. Build the Interactive adapter executable by running adadmin.
   a. Select the option "Relink Applications Programs".
   b. Enter the following values at the corresponding prompts:

Prompt	Value to Enter
Enter list of products to link ('all' for all products) [all]:	xdp
Generate specific executables for each selected product [No]?	yes
Enter executables to relink, or enter 'all' [all]:	XDPNTRCT

Successful completion of above step will build the required executable in $XDP_TOP/bin. In case of errors, contact your Oracle Support representative.

Oracle CRL Financials

Enabled Assets

Perform the following steps before using CRL functionality.

Post-Installation

Setup CRL Profile (Required)

If you are licensed to use CRL-FA and want to use the CRL Fixed Assets functionality then run the following script as the apps user.

```
$CUA_TOP/patch/115/sql/FACSTPRF.sql
```

Pre-Upgrade

If you already have CRL Enabled Assets installed and wish to upgrade to Oracle Assets/CRL 11*i*, please do the following to ensure successful upgrade.

1. Disable any CRL Enabled Assets responsibilities and Oracle Asset responsibilities. (Required)
2. Check CRL responsibility being used.

> **Caution:** Oracle Assets 11*i* installs CRL with a seeded responsibility as CRL Assets Manager. Make sure that a responsibility with this name does not exist on your system at the time of upgrade.

Post-Upgrade

1. Setup CRL Profile (Required)

   If you are licensed to use CRL-FA and want to use the CRL Fixed Assets functionality then run the following script as the apps user.

   ```
   $CUA_TOP /patch/115/sql/FACSTPRF.sql
   ```

2. Setup Key Flexfields (Required)

   The new key flexfields need to be configured to match the old flexfield configuration. The following table lists the new key flexfields and the corresponding old key flexfields that need to be re-configured.

Old Key Flexfield Name	Old Application Name	New Key Flexfield Name	New Application Name
Group Asset	Oracle CRL Enabled Assets	Group Assets	Oracle Assets
Super Group	Oracle CRL Enabled Assets	Super Group	Oracle Assets

3. Setup Descriptive Flexfields (Required)

   The new descriptive flexfields need to be configured to match the old flexfield configuration. The following table lists the new descriptive flexfields and the corresponding old descriptive flexfields that need to be re-configured.

Old Desc Flexfield Name	Old Application Name	New Desc Flexfield Name	New Application Name
IFA_HIERARCHY_ DESC_FLEX	Oracle CRL Enabled Assets	FA_HIERARCHY_ DESC_FLEX	Oracle Assets
IFA_HIERARCHY_ RULE_DESC_FLEX	Oracle CRL Enabled Assets	FA_HIERARCHY_ RULE_DESC_FLEX	Oracle Assets

4. Drop obsolete tables ONLY after confirming that your data was successfully upgraded. (Optional)

Use the following script to delete these tables.

```
$CUA_TOP/patch/115/sql/FACDTAB.sql
```

- ifa_books_groups
- ifa_group_assets
- ifa_group_asset_default
- ifa_group_asset_rules
- ifa_group_deprn_detail
- ifa_group_deprn_rates
- ifa_group_deprn_summary
- ifa_super_groups
- ifa_super_group_rules
- ifa_mass_external_transfers
- ifa_mass_ext_retirements
- ifa_ext_inv_retirements
- ifa_mass_ext_ret_excepts
- ifa_parallel_workers
- ifa_system_controls
- ifa_asset_hierarchy
- ifa_asset_hierarchy_values
- ifa_hierarchy_rule_set

- ifa_hierarchy_rule_details
- ifa_exclude_hierarchy_levels
- ifa_asset_hierarchy_purpose
- ifa_mass_update_batch_headers
- ifa_mass_update_batch_details
- ifa_hr_retirement_headers
- ifa_hr_retirement_details
- ifa_hierarchy_controls
- ifa_hierarchy_distributions
- ifa_life_derivation_info
- ifa_mc_group_deprn_summary
- ifa_mc_group_deprn_details
- ifa_mc_books_groups
- ifa_mass_additions

5. Drop obsolete views ONLY after confirming that your data was successfully upgraded. (Optional)

   Use the following script to delete these views.

   `$CUA_TOP/patch/115/sql/FACDVWS.sql`

   - IFA_ASSET_HIERARCHY_DFV
   - IFA_ASSET_HIERARCHY_PURPOSE_V
   - IFA_ASSET_HIERARCHY_V
   - IFA_ASSET_HIERARCHY_VALUES_V
   - IFA_ASSET_HRCHY_DETAILS_V
   - IFA_BOOKS_GROUPS_V
   - IFA_BOOKS_GROUPS_V1
   - IFA_CHILD_ASSETS_V
   - IFA_CURR_DEPRN_ADJ_V
   - IFA_DEPRN_GROUPS_V

- IFA_DEPRN_GROUPS_V1
- IFA_GROUP_ASSETS_KFV
- IFA_GROUP_COST_INQUIRY_V
- IFA_HIERARCHY_BATCH_DETAILS_V
- IFA_HIERARCHY_BATCH_HEADER_V
- IFA_HIERARCHY_DISTRIBUTIONS_V
- IFA_HIERARCHY_RULE_DETAILS_V
- IFA_HR_BATCH_DIST_NEW_V
- IFA_HR_BATCH_DIST_OLD_V
- IFA_HR_RETIREMENT_DETAILS_V
- IFA_HR_RETIREMENT_HEADERS_V
- IFA_MASS_EXTERNAL_TRANSFERS_V
- IFA_MASS_EXT_RETIREMENTS_V
- IFA_MUPD_V
- IFA_SOURCE_LINES_V
- IFA_SUPER_GROUPS_KFV
- IFA_LIFES_V

6. Reduce obsolete columns after confirming your upgrade. (Optional)

Use the following script to delete these columns.

`$CUA_TOP/patch/115/sql/FACDCOL.sql`

- fa_mass_additions.group_asset_id
- fa_mass_additions.ifa_parent_hierarchy_id
- fa_category_book_defaults.ifa_life_end_date
- fa_category_book_defaults.ifa_rule_set_id
- fa_books.group_asset_id

7. Drop obsolete triggers after confirming your upgrade. (Optional)

Use the following script to delete these triggers:

```
$CUA_TOP/patch/115/sql/FACDTRG.sql
```

- IFA_ADDITIONS_HR_ARD
- IFA_ADJUST_UNITS_ARI
- IFA_ADJUST_UNITS_ARU
- IFA_ASSET_INVOICES_BRI
- IFA_BOOKS_GROUPS_BRI
- IFA_BOOK_CONTROLS_BRU
- IFA_MASS_ADDITIONS_ARU
- IFA_RETIREMENTS_BRU
- IFA_RETIRMENTS_ARU
- IFA_TRANSACTION_HEADERS_ARI2
- IFA_TRANSACTION_HEADERS_ARI3
- IFA_TRANSACTION_HEADERS_HR_BRI

8. Drop obsolete packages after confirming your upgrade. (Optional)

Use the following script to delete these packages:

```
$CUA_TOP/patch/115/sql/FACDPKG.sql
```

- IFADEPR
- IFA_ASSET_APIS
- IFA_ASSET_WB_APIS_PKG
- IFA_BOOKS_GROUPS_PKG
- IFA_CALC_NBV_PKG
- IFA_CLIENT_EXTENSION
- IFA_DERIVE_ASSET_ATTR_PKG
- IFA_EXT_TRANSFERS_PKG
- IFA_FLEX_BLD_PKG
- IFA_FLEX_BUILD_PKG
- IFA_GROUP_RET_ADJ_PKG

- IFA_HIERARCHY_DISTRIBUTION_PKG
- IFA_HIERARCHY_PKG
- IFA_HIERARCHY_PURPOSE_PKG
- IFA_HIERARCHY_RULE_DETAILS_PKG
- IFA_HIERARCHY_VALUES_PKG
- IFA_HR_REINSTATEMENTS_PKG
- IFA_HR_RETIREMENTS_PKG
- IFA_INVOICE_TRANSACTIONS_PKG
- IFA_MASS_EXT_RET_PKG
- IFA_MASS_UPDATE1_PKG
- IFA_MASS_UPDATE2_PKG
- IFA_RECLASS_PKG
- IFA_SYSTEM_CONTROL_PKG

9. Refer to the *Oracle CRL Financials Implementation Guide* on the Oracle Applications, Release 11*i* Documentation CD-ROM for additional post-installation and implementation tasks. (Required)

CRL Enabled Projects

Perform the following post installation/upgrade steps to use the CRL Projects functionality.

1. Setup CRL Profile

   If you are licensed to use CRL-PROJECTS and want to use the CRL Projects functionality then set the profile "PA:Licensed to use CRL projects" to **Yes** at the site level.

2. The CRL Projects Manager responsibility must be enabled by setting the effective end date to **null**.

3. The two client extensions packages for CRL - Project customers must be modified to uncomment the default CRL Projects functionality and applied. The packages are:

   - PA_CLIENT_EXTN_GROUPING (IPAGCEB.pls)
   - PA_CLIENT_EXTN_GEN_ASSET_LINES (IPAGALCB.pls)

4. Setup Descriptive Flexfields (Required)

   Descriptive flexfields must be configured as documented in *Oracle CRL-Financials Enabled Projects Concepts and Procedures, Release 11i*. This setup must be done manually.

   The following table lists the new key flexfields and the corresponding old key flexfields that need to be re-configured.

Descriptive Flexfield Name	Columns	Value Set Used	Application Name
Expenditure Items	Attribute8 Attribute9 Attribute10	Depends on the naming convention. Refer to the *Oracle CRL Projects Concepts and Procedures* manual.	Oracle Projects
Expenditure Types Desc flex	Attribute9 Attribute10	PA_SRS_YES_NO_LOVPA_SRS_ YES_NO_LOV	Oracle Projects
Projects	Attribute10	PA_SRS_YES_NO_LOV	Oracle Projects
Tasks	Attribute9	PA_SRS_YES_NO_LOV	Oracle Projects

   - For new CRL customers, make sure these descriptive flex field attributes are not used for other purposes. In addition, CRL post-installation/upgrade mandatory patch #1238551 needs to be applied.
   - Existing CRL Projects users must verify with these decsriptive flexfields for accuracy.

5. Refer to the *Oracle CRL Financials Implementation Guide* on the Oracle Applications, Release 11*i* Documentation CD-ROM for additional post-installation and implementation tasks. (Required)