Siebel Incentive Compensation Management Installation Guide for UNIX

Version 7.8.2, Rev. A

January 2006

Siebel Systems, Inc., 2207 Bridgepointe Parkway, San Mateo, CA 94404

Printed in the United States of America

Contents

Chapter 4: Siebel Incentive Compensation Management Installation

Chapter 5: Postinstallation Tasks

1 What's New in This Release

What's New in Siebel Incentive Compensation Management Installation Guide for UNIX, Version 7.8.2, Rev. A

Table 1 lists changes described in this version of the documentation to support release 7.8.2 of the software.

Table 1. New Product Features in Siebel Incentive Compensation Management Installation Guide for UNIX, Version 7.8.2, Rev. A

Topic	Description
"Installing WebSphere" on page 25	Instructions for installing WebSphere have been updated to accurately reflect the latest JDK fix.
"Configuring ICM for DB2 on UNIX" on page 38	Instructions for configuring ICM for DB2 on UNIX have been added.
"Setting the LOCKTIMEOUT Parameter for DB2 on UNIX" on page 38	Instructions for configuring the DB2 database have been expanded to set the LOCKTIMEOUT parameter and to add 16 KB table spaces and buffer pool.
"Deploying ICM on an Application Server" on page 44	Instructions for deploying ICM have been updated to specify correct directory paths.
"Configuring Environment Settings for LDAP Authentication" on page 56 "Configuring LDAP Authentication for JBoss" on page 61 "Configuring LDAP Authentication for WebSphere" on page 64	LDAP authentication topics are reorganized under JBoss and Websphere authentication headings.
"Configuring Alternate Authentication Modes for JBoss" on page 60	A procedure has been added with common steps for configuring non-default authentication modes for JBoss.
"Configuring Alternate Authentication Modes for WebSphere" on page 63	A procedure has been added with common steps for configuring non-default authentication modes for WebSphere.
"Upgrading the ICM Database Schemas" on page 68	Upgrade instructions for the ICM database schemas have been reorganized to reflect the addition of SERVICE_TYPE table upgrade scripts.
"Upgrading the ICM Transaction Database Schema for DB2" on page 69	Upgrade procedure for the ICM DB2 database schema has been revised and expanded to set the LOCKTIMEOUT parameter.

Table 1. New Product Features in Siebel Incentive Compensation Management Installation Guide for UNIX, Version 7.8.2, Rev. A

Topic	Description
"Process of Setting Up Informatica PowerCenter with ICM" on page 74	Process description for setting up Informatica PowerCenter has been revised to mention the installation of PowerCenter components.
"Importing the ICM Repository into PowerCenter" on page 76	The procedure for importing the ICM repository into Informatica PowerCenter has been updated to specify the correct directory paths and repository file.
"Configuring the PowerCenter Server for ICM" on page 78	Additional steps are added for completing the Informatica PowerCenter server configuration.
"Installing the ICM and Actuate Security Extension for UNIX" on page 116	Procedures for installing the Actuate Security Extension have been updated to specify the correct directory paths and files.

What's New in Siebel Incentive Compensation Management Installation Guide for UNIX, Version 7.8.2

Table 2 lists changes described in this version of the documentation to support release 7.8.2 of the software.

Table 2. New Product Features in Siebel Incentive Compensation Management Installation Guide for UNIX, Version 7.8.2

Topic	Description
"Process of Performing a Quick Installation" on page 16	A process for performing a quick install of ICM has been added to the guide.
"Installing Oracle" on page 20	The procedure for installing Oracle has been expanded to specify settings needed to make the Oracle database ICM compatible.
"Installing DB2" on page 23	The procedure for installing DB2 has been expanded to specify settings needed to make the DB2 database ICM compatible.
"Installing jchardet" on page 30	A procedure for installing the third-party jchardet utility has been added to the guide.
"Configuring Number, Currency, Date, and Time Formats" on page 41	A procedure for customizing number, currency, date, and time formats is added to the guide.
"Configuring Apache as HTTP Server for JBoss" on page 54	A procedure is added for configuring the system to use Apache as the HTTP Server in place of Tomcat.

Table 2. New Product Features in Siebel Incentive Compensation Management Installation Guide for UNIX, Version 7.8.2

Topic	Description
Appendix A, "Upgrading from ICM 7.8 to 7.8.2"	Instructions for upgrading from the previous version of ICM to the current version of ICM have been updated for the latest release.
Appendix B, "Siebel ICM and Informatica PowerCenter"	Instructions for installing Informatica have been expanded for Unicode users.

What's New in Siebel Incentive Compensation Management Installation Guide for UNIX, Version 7.8 Rev. A

Table 3 lists changes described in this version of the documentation to support release 7.8 of the software.

Table 3. New Product Features in Siebel Incentive Compensation Management Installation Guide for UNIX, Version ICM 7.8, Rev. A

Topic	Description
"About ICM Architecture" on page 14 "Installing the Database Application" on page 20	Support for the DB2 database platform
"Installing the Application Server" on page 24	Support for the WebSphere application server platform
"Configuring the ICM Properties Files" on page 36	Listings of previously undocumented properties
"Configuring the ICM Properties Files" on page 36	Newly added descriptions of properties
"Deploying ICM on an Application Server" on page 44	Changes to command names
Chapter 5, "Postinstallation Tasks"	Addition of WebSphere, IIS, authentication, and other post-installation procedures
"Configuring ICM Authentication Modes for JBoss" on page 59	Support for a default security system
"Configuring ICM Authentication Modes for WebSphere" on page 62	Support for WebSphere authentication
Appendix A, "Upgrading from ICM 7.8 to 7.8.2"	Instructions for upgrading from the previous version of ICM to the current version of ICM
Appendix B, "Siebel ICM and Informatica PowerCenter"	Support for Informatica

Table 3. New Product Features in Siebel Incentive Compensation Management Installation Guide for UNIX, Version ICM 7.8, Rev. A

Topic	Description
Appendix C, "Setting Up ICM Distributed Processing"	New instructions for setting up distributed processing
Appendix D, "Siebel ICM Report Utility"	Support for a reporting module, as a stand-alone utility or integrated with ICM
Appendix E, "Configuring Actuate iServer for Siebel ICM"	Support for configuring Actuate Server for Siebel ICM
Appendix F, "Target Commands Reference"	New Ant targets

2 Overview of Siebel Incentive Compensation Management

This chapter provides an overview of the Siebel Incentive Compensation Management application. The chapter consists of the following sections:

Who Should Read the ICM Installation Guide

This guide is for system administrators and database administrators responsible for the installation, configuration, and maintenance of Siebel Incentive Compensation Management host servers and databases. This guide contains technical information on the planning and implementation of Siebel Incentive Compensation Management. It does *not* address advanced database configuration, Siebel Incentive Compensation Management customization, or user-level features.

This document assumes a basic familiarity with Solaris or AIX, and with Oracle administration or DB2 administration. While important commands and steps related to Siebel Incentive Compensation Management installation are described in detail, some commands may not be familiar to junior administrators. Siebel Incentive Compensation Management contains additional database and server components that require advanced technical setup.

About Siebel ICM

Siebel Incentive Compensation Management is used for designing, managing, and reporting incentive compensation plans. Siebel Incentive Compensation Management is typically used by financial administrators and corporate managers to provide a company's employees with bonuses, commissions, and other incentives based on individual or group performance benchmarks. Siebel Incentive Compensation Management can also manage incentive plans for channel partners, value-added resellers, or any group or person for whom commissions and rewards are an essential part of the business relationship.

You can deploy ICM on one application server or on multiple application servers. A *stand-alone configuration* is a configuration in which all components of Siebel ICM are installed on one application server. Stand-alone configurations work best for small groups or companies. *Distributed processing* describes a configuration in which components of Siebel ICM are installed on multiple application servers to distribute the services processing. Distributed processing configurations work best for large groups or companies.

ICM is comprised of three self-contained tiers of integrated components. In a stand-alone configuration, all three tiers are hosted on the same server. As needs grow, any tier can be migrated to a dedicated server or upgraded without affecting other aspects of the system. In a distributed processing configuration, one or more tiers can have their own dedicated servers. For more information about tiers, see "About ICM Architecture" on page 14.

Siebel Incentive Compensation Management uses the Java 2 Enterprise Edition (J2EE) platform and Enterprise JavaBeans (EJB) to promote maximum flexibility during initial rollout and upgrading. Access to data, reports, and metrics is through a zero-footprint, Web browser-based client. No additional client-side software is required.

ICM runs on a variety of platforms. ICM supports UNIX and Windows operating systems; JBoss and WebSphere application servers; and Oracle, SQL Server, and DB2 databases. Some installation steps differ depending on your company's OS, application server, and database.

About ICM Architecture

This section describes Siebel ICM's Web Tier, Application Tier, and Data Tier.

Web Tier

The *Web Tier* is tier of Siebel ICM that serves the processed information from the data and application tiers and sends it to the client software on the end user's desktop. The Web tier serves as the portal to Siebel Incentive Compensation Management. It gives end users access to the ICM application, and allows them to obtain and add information through Microsoft Internet Explorer.

The Web Tier provides access to ICM through a combination of a Web server, Java servlets, and Java Server Pages, or JSPs, which are defined as follows:

- A *Java Server Page (JSP)* is a technology that uses Java to modify Web pages before they are sent to a user.
- A *servlet* is a small, Java-based program that configures the layout or content of a JSP.

The Web Tier uses the Tomcat Web server to process JSPs and to send formatted data to client browsers.

Application Tier

The *Application Tier* is the logic tier of ICM. It contains the software that processes business logic and transaction rules. It can interface with multiple external legacy systems, including Enterprise Resource Planning (ERP) and Contact Relationship Management (CRM) applications. The Application Tier makes up the bulk of Siebel Incentive Compensation Management components. It includes the analytical engine, security features, user management interface, transaction rules engine, and other components that process the raw data and statistics. This tier uses the JBoss or WebSphere application server. JBoss uses Enterprise JavaBeans to process Siebel Incentive Compensation Management's business logic and to interact with legacy systems.

Data Tier

The *Data Tier* contains Siebel Incentive Compensation Management's database and all related raw data. It contains Siebel Incentive Compensation Management's transactional and analytics databases. The data schema operates under an Oracle, SQL Server, or DB2 relational database instance. These components are *not* bundled with Siebel Incentive Compensation Management and must be procured and licensed separately.

About Other ICM Documents

In addition to the Siebel Incentive Compensation Management's online help, the product is accompanied by the following guides, which describe Siebel Incentive Compensation Management from implementation to every day use:

- ***Siebel Incentive Compensation Management Configuration Guide.*** This guide provides directions for system administrators or financial planners responsible for configuring the Siebel Incentive Compensation Management environment for a group or company after a successful installation.
- ***Siebel Incentive Compensation Management Administration Guide.*** This guide details Siebel Incentive Compensation Management's essential functionality. It is designed for financial planners, payroll staff, or anyone needing client-level access to Siebel Incentive Compensation Management.

Processes for ICM Installation

There are two sequential processes. First you install a database, an application server, and third-party applications that are required for ICM. Then you install and configure the ICM application itself. The steps of these processes are listed in the following sections:

1 "Process of Meeting the Requirements for ICM Installation" on page 19

2 "Process of Installing and Configuring ICM" on page 35

Process of Performing a Quick Installation

You can install ICM with a variety of options and configurations, as described in the chapters that follow. This topic describes a simplified version of the installation process, which produces a quick, basic ICM deployment. The process assumes that you are performing a new installation and that you will run ICM on an Oracle or SQL Server database platform.

NOTE: ICM requires several third-party applications, including a database and an application server. ICM supports specific versions of these applications. For types and versions of third-party software supported by your ICM application, see *System Requirements and Supported Platforms* on SupportWeb.

1. Identify the type and version of the database your ICM instance will use, and confirm that it is supported.
2. Install the application server and JDK.
   a. Install JBoss and Tomcat. See "Installing JBoss and Tomcat" on page 25.

      The directory in which you install JBoss and Tomcat will become the root directory of your ICM installation. This directory is known throughout this guide as <DEPLOYMENT>.

   b. Install the supported Java Developers Kit. See "Installing Java Developers Kit for JBoss" on page 28.

      The directory in which you install the Java Developers Kit is known throughout this guide as <JAVA_HOME>.
3. Download the ICM distribution files to a staging directory. See "Unpacking the ICM Software to a Staging Directory" on page 29.

   The directory where you configure the ICM files before deployment is known throughout this guide as <STAGING>.
4. Install third-party applications.
   a. Install Mozilla Rhino. See "Installing Mozilla Rhino" on page 29.
   b. Install iText. See "Installing iText" on page 29.
   c. Install jchardet. See "Installing jchardet" on page 30.
   d. Install Apache Ant. See "Installing Apache Ant" on page 31.

      The directory where you install Apache Ant is known throughout this guide as <ANT>.
5. Install the appropriate JDBC driver for your database.
   - For SQL Server, the JDBC driver comes with your ICM system, so no action is necessary.
   - For Oracle, see "Installing JDBC Drivers for Oracle" on page 30.
6. Configure the ICM properties.
   a. Edit the ICM deployment properties. See "Configuring ICM Deployment Properties" on page 37.

b Edit the application server properties for JBoss. See "Configuring ICM for JBoss" on page 39.

Add the appropriate values for deploy.appserver.root(<DEPLOYMENT>) and deploy.java.home(<JAVA_HOME>).

c Edit the database properties.

- For SQL Server, add the appropriate values for db.transactionDB.sa.user, db.transactionDB.sa.password, db.analyticsDB.sa.user, and db.analyticsDB.sa.password. See "Configuring ICM for SQL Server" on page 38.
- For Oracle, see "Configuring ICM for Oracle" on page 38.

7 Deploy and launch your ICM application.

a Start the database server.

b At a command prompt, navigate to the <STAGING>\deploy directory and deploy the ICM application by entering the following command:

```
ant deploy
```

c Start the application server. See "Starting the JBoss Application Server" on page 46.

d At a command prompt, navigate to the <STAGING>\deploy directory and populate the ICM dictionary and system by entering the following command:

```
ant populate-all
```

e After the system displays a BUILD SUCCESSFUL message, the installation is complete and you can access the application. See "Creating Your Own Data in ICM" on page 48.

3 Meeting the Requirements for Installing Siebel ICM

Before you can install Siebel ICM, you have to meet certain requirements for applications on which ICM depends to be installed on your company's system.

This chapter describes the requirements for beginning an ICM installation. It includes the following topics:

Process of Meeting the Requirements for ICM Installation

Siebel ICM supports specific versions of third-party applications. For a definitive list of supported third-party products, see *System Requirements and Supported Platforms*.

NOTE: It is recommended that Siebel Incentive Compensation Management and its adjunctive applications be installed on a disk partition separate from the partition that hosts the operating system and system logs. It is not necessary to place tiers on individual partitions.

The following process summarizes the steps for installing Siebel Incentive Compensation Management, in the order in which they are performed.

1. "Installing the Database Application" on page 20

   Follow *only* the procedures for the database you use with ICM.

   - "Installing Oracle" on page 20
   - "Populating the Product User Profile for Oracle" on page 23
   - "Installing SQL Server" on page 23
   - "Installing DB2" on page 23

2. "Installing the Application Server" on page 24

   Follow the procedures for JBoss and Tomcat, even if you use WebSphere for the application server. In this case, also follow the procedures for WebSphere.

   a. "Installing JBoss and Tomcat" on page 25

b “Installing WebSphere” on page 25

3 “Downloading and Installing Third-Party Software for Siebel ICM” on page 28

Follow the procedures for installing server- and database-related software *only* if they are appropriate to your configuration.

a “Installing Java Developers Kit for JBoss” on page 28

b “Unpacking the ICM Software to a Staging Directory” on page 29

c “Installing Mozilla Rhino” on page 29

d “Installing iText” on page 29

e “Installing jchardet” on page 30

f “Installing JDBC Drivers for Oracle” on page 30

g “Installing JDBC Drivers for DB2” on page 31

h “Installing Apache Ant” on page 31

i “Installing the X11 Server” on page 32

j “Installing hsqldb.jar for WebSphere” on page 32

k “Downloading and Installing the Report Utility” on page 33

This procedure is required *only* if you use ICM’s Reports module.

4 “Creating a Siebel ICM User” on page 33

This procedure is optional but recommended.

Installing the Database Application

Siebel Incentive Compensation Management retrieves raw data from a relational database that makes up the Data Tier. Before installing Siebel ICM, you must implement and configure the database application. Siebel ICM supports the Oracle, SQL Server, and DB2 database applications. For information about specific supported versions of these applications, see *System Requirements and Supported Platforms* on Siebel SupportWeb.

The following topics describe database application installation. Refer to the procedures that are appropriate to the database you install:

- “Installing Oracle” on page 20
- “Populating the Product User Profile for Oracle” on page 23
- “Installing SQL Server” on page 23
- “Installing DB2” on page 23

Installing Oracle

If you use Oracle as the Siebel ICM database, you must install Oracle before installing ICM.

This topic specifies field values and option selections that make the Oracle database compatible with ICM. For detailed instructions on installing Oracle and reference information about values and options, consult the Oracle documentation.

NOTE: If you install Oracle on one machine and you plan to install the ICM software on another machine, be sure to install the Oracle Client Tools on the machine that have the ICM <STAGING> directory. This installation allows you to run the database configuration targets in the <STAGING> directory against a database on another machine.

This task is a step in "Process of Meeting the Requirements for ICM Installation" on page 19.

To install Oracle

1. Launch the Oracle Database Configuration Assistant and create a new database.
2. In the Configuration Assistant screens, complete the fields and selection options. Selections and values for the items that affect ICM compatibility are described in the following table.

Item	Comments
SID	Must be 8 characters or less. It is recommended that the SID match the first part of the Global Database Name.
Shared Server Mode	Recommended with the TCP option for the ICM database.
Memory settings	To obtain custom Pool and Cache memory settings for your installation, contact your Siebel Technical Support representative.
Initialization Parameters	See "Additional Initialization Parameters" on page 22.
Database Character Set	For Western European languages, select AL32UTF8. For Asian languages, you can use UTF8 or AL16UTF16. For more information about these options, see "Character Sets in Oracle Databases" on page 23. **CAUTION:** You cannot change the Database Character Set after you have created the database.
National Character Set	For Western European languages, select UTF8. For Asian languages, you can use UTF8 or AL16UT16. For more information about these options, see "Character Sets in Oracle Databases" on page 23.
Block Size	A minimum block size of 8K is recommended when running a database character set of UTF8 or UTF16. **CAUTION:** If the block size is too small, your system may experience excessive disc I/O operations and errors during ICM schema index creation. If the block size is too large, your system may experience inefficient memory use and resource contention. To identify the best settings for your installation, contact your Siebel Technical Support representative.

Item	Comments
Tablespaces	Under Datafiles, create a tablespace for each tablespace name specified in the ICM <STAGING>\deploy\oracle\db.properties configuration file. Suggestions are DATA, INDEXES, DATA_ANALYTICS, and INDEXES_ANALYTICS. Configure these tablespaces to accommodate the projected data size and growth rate of your ICM installation. If you are unsure what settings to use, you can specify 700MB for each of the four tablespaces and change the settings later if necessary. **NOTE:** If you specify 700MB for all four tablespaces, be sure that you have enough disk space to accommodate four 700MB table spaces (2.8GB).
Tablespace Storage	It is recommended that you select the following options: ■ Locally managed ■ Automatic Allocation ■ Automatic ■ Yes - Generates redo logs and recoverable
Creation Options	It is recommended that you select the following options: ■ Create Database ■ Save as Database Template ■ Generate Database Creation Scripts Selecting the last two items makes it easier to recreate an ICM-compatible database, if this becomes necessary.

Additional Initialization Parameters

The parameter values listed in Table 4 are recommended for ICM compatibility.

Table 4. Initialization Parameters

Parameter	Value
nls_length_semantics	CHAR
nls_nchar_conv_excp	FALSE for a new production system TRUE for a staging system, test system, or a system migrated from an earlier version of Siebel ICM
nls_language	AMERICAN (default). Consult the Oracle documentation for other language values.
nls_territory	AMERICA (default). Consult the Oracle documentation for other territory values.

Character Sets in Oracle Databases

For single-byte character sets such as ISO8859-1 Latin, UTF8 uses disk space more efficiently than UTF16. However, the jdbc driver must convert UTF8 data to UTF16 (the Java standard) at run time, and therefore yields slower performance than UTF16. Most companies with ICM instances that use Western European character sets select AL32UTF8 to conserve disk space.

For Asian languages, UTF8 will support all double-byte character needs and is recommended, but you can also experiment with UTF16 values. If your system will store mostly double-byte characters (for example, Simplified Chinese, Korean, or Russian), UTF-16 will provide more efficient storage than UTF8. If you are not sure which character set is appropriate, then use UTF-8.

Populating the Product User Profile for Oracle

This procedure applies to Oracle installations only.

After the Oracle instance is installed and verified, you must run a script before moving on to install the Siebel ICM software. This script populates the Product User Profile information required by SQL*Plus.

This task is a step in "Process of Meeting the Requirements for ICM Installation" on page 19.

To populate the Product User Profile for Oracle

1. On the transaction database host, launch SQL*Plus.
2. Log in with the SYSTEM user name and password.
3. Run <ORACLE_HOME>/sqlplus/admin/pupbld.sql. Example:

   ```
   sqlplus system/manager @<ORACLE_HOME>/sqlplus/admin/pupbld.sql
   ```

Installing SQL Server

If you use SQL Server as the Siebel ICM database, you must install SQL Server 2000 before installing ICM.

This task is a step in "Process of Meeting the Requirements for ICM Installation" on page 19.

To install SQL Server

- For instructions on installing SQL Server, consult the SQL Server documentation.

Installing DB2

If you use DB2 as the Siebel ICM database, you must install DB2 before installing ICM.

This topic specifies field values and option selections that make the DB2 database compatible with ICM. For detailed instructions on installing DB2, and for reference information about values and options, consult the DB2 documentation.

NOTE: If you install DB2 on one machine and you plan to install the ICM software on another machine, be sure to install the DB2 Client Tools on the machine that have the ICM <STAGING> directory. This installation allows you to run the database configuration targets in the <STAGING> directory against a database on another machine.

This task is a step in "Process of Meeting the Requirements for ICM Installation" on page 19.

To install DB2

1 Launch the DB2 Create Database Wizard and create a new database.

2 In the Create Database Wizard screens, complete the fields and selection options.

Selections and required values for the items that affect ICM compatibility are described in the following table.

Items	Values
Code Set	UTF-8
Collating Sequence	System

3 Add the following lines to the .profile file for the user account that deploys ICM:

```
If [ -f<DB2_HOME>/sqllib/db2profile ]; then
. <DB2_HOME>/sqllib/db2profile
fi
```

The user account that deploys ICM runs DB2 commands to create the DB2 schema and stored procedures. Therefore, this user needs to have db2profile environment settings.

Installing the Application Server

Before installing Siebel ICM, you must install the application server on which ICM runs. Siebel ICM supports the JBoss and WebSphere application servers. For information about specific supported versions of these applications, see *System Requirements and Supported Platforms* on SupportWeb.

Refer to whichever of the following procedures that is appropriate to your application server installation:

- "Installing JBoss and Tomcat" on page 25
- "Installing WebSphere" on page 25

Installing JBoss and Tomcat

This procedure applies to all installations.

CAUTION: If you choose JBoss as the application server, you *must* install the integrated JBoss and Tomcat. Even if the installation is only used as a processor node for distributed processing, Tomcat is still required.

JBoss is the default application server for Siebel ICM. JBoss is an open-source, Java 2 Enterprise Edition (J2EE) Web operating system that provides a link to legacy applications and systems through Enterprise JavaBeans. Tomcat is an open-source container for Java servlets and Java Server Pages (JSPs).

This task is a step in "Process of Meeting the Requirements for ICM Installation" on page 19.

To install JBoss and Tomcat

1 Download JBoss/Tomcat from the following location:

```
http://prdownloads.sourceforge.net/bulldog/jboss3.0.3_tomcat4.1.12-
MOD.zip?download
```

2 Unzip the jboss3.0.3_tomcat4.1.12-MOD.zip file to a location of your choice; for example:

```
/home/siebelicm/
```

NOTE: This location is referred to throughout this guide as the <DEPLOYMENT> directory.

Installing WebSphere

This procedure applies to WebSphere installations only.

If you use WebSphere as the Siebel ICM application server, you must install WebSphere and its fixes before installing ICM.

It is recommended while you are installing WebSphere to note all the port numbers, cell name, node name, and server name because you need them when you deploy Siebel ICM. Note that cell, node, and server names are case sensitive.

NOTE: Siebel ICM uses WebSphere with the AIX platform only, and not with other flavors of UNIX.

This task is a step in "Process of Meeting the Requirements for ICM Installation" on page 19.

To install WebSphere on AIX

1 Log on as the root user.

CAUTION: If you install WebSphere as a nonroot user, you may experience permission problems when setting up and configuring ICM on WebSphere. Please use the "root" privilege for all WebSphere-related operations including installing WebSphere, deploying ICM on WebSphere, and running WebSphere.

2 Verify that no JAVA installation is set up in the system path.

3 Follow the installation instructions at the following Web site:

```
http://publib.boulder.ibm.com/infocenter/ws51help/index.jsp?topic=/
com.ibm.websphere.base.doc/info/aes/ae/tins_install.html
```

4 Make sure you meet all the prerequisites listed at the following Web site:

```
http://www-306.ibm.com/software/webservers/appserv/doc/v50/prereqs/was_v51.htm
```

   a Check the operating system version by running the following command:

   ```
   oslevel -r
   ```

   b Check whether APAR IY44183 is installed by running the following command:

   ```
   instfix -i |grep IY44183
   ```

   If the command runs correctly, the system returns the following message:

   ```
   All filesets for IY44183 were found.
   ```

   c Check whether the XLC 6.0 RTE is installed by running the following command:

   ```
   instfix -iv |grep xlC.rte:6.0.0.0
   ```

   If the command runs correctly, the system returns the following message:

   ```
   Fileset xlC.rte:6.0.0.0 is applied on the system.
   ```

5 Remove old versions of the http server.

6 Perform the following steps:

```
echo "Creating Group mqm..."

mkgroup -'A' mqm

echo "Creating Group mqbrkrs..."

mkgroup -'A' mqbrkrs

echo "Creating user mqm as member of mqm group..."

mkuser pgrp=mqm mqm

echo "Adding root to mqm and mqbrkrs groups"

chgrpmem -m + root mqm

chgrpmem -m + root mqbrkrs

echo "Disabling websm Which hogs up port 9090..."

/usr/websm/bin/wsmserver -disable
```

7 Log off and log back on as the root user.

8 Start the WebSphere installation from the installation media (usually /cdrom/aix/install) and follow the prompts.

It is recommended that you install WebSphere in the following directory:

`<DEPLOYMENT>/WebSphere/AppServer/`

9 Install the WebSphere Cumulative Fix Pack by performing the following steps:

a Go to the following IBM Web site:

`ftp://ftp.software.ibm.com/software/websphere/appserv/support/fixpacks/was51/cumulative/cf5101/`

NOTE: The download location for fix packs is subject to change by IBM.

b Download the cumulative fix for your OS.

c From the unzipped directory, run the following command:

`java -version`

It should not point to any JDK other than the WebSphere-embedded JDK. If it points to any other JDK, remove it from the System PATH parameter.

d Follow the instructions on the Web site to install the cumulative fix.

10 Install the IBM JDK fix by performing the following steps:

a Go to the IBM WebSphere Support Web site:

`http://www-1.ibm.com/support/docview.wss?uid=swg24008901`

b Follow the installation instructions on the displayed Web page to download and install the JDK that is correct for your operating system and application server.

c Confirm that the correct JDK was installed.

For a definitive list of JDKs that are correct for supported operating systems and application servers, see *System Requirements and Supported Platforms*.

11 Install the Interim fix for MQ by performing the following steps:

a Go to the following Web site:

`https://www6.software.ibm.com/dl/wsmqcsd/wsmqcsd-p`

b From the displayed Web page, download and install the following:

WebSphere Embedded Messaging interim fixes for WebSphere Application Server V5.1

c Apply the fix that is needed for ICM.

The fix is displayed on the Web page. For the APAR number that is appropriate for your ICM installation, see *System Requirements and Supported Platforms*.

NOTE: The MQ fix APAR number and other installation links that are provided by IBM are subject to change by IBM.

12 Add the following to the System Path:

```
<DEPLOYMENT>\bin
```

Downloading and Installing Third-Party Software for Siebel ICM

Siebel ICM requires several third-party software packages to provide or support some of its functionality. You must obtain some of these programs from their respective manufacturers. This section explains how to download and install third-party software required by Siebel ICM. This section contains the following subsections:

You install the third-party software in the order listed in this section.

Installing Java Developers Kit for JBoss

This procedure applies to JBoss installations only.

Siebel ICM requires the Java Developers Kit (JDK) to operate. The JDK provides the run-time Java Virtual Machine used by Siebel ICM. The JDK also provides the compiler used to compile the Java Server Pages (JSPs) that generate the user interface.

The instructions that follow for installing Java Developers Kit apply *only* if you use JBoss and Tomcat. If you use WebSphere, use the JDK and its associated eFix that ship with WebSphere. For information on installing WebSphere, see "Installing WebSphere" on page 25.

NOTE: The JDK installation directory is referred to throughout this guide as <JAVA_HOME>. The deploy.java.home property sets the location of the Java Developers Kit.

This task is a step in "Process of Meeting the Requirements for ICM Installation" on page 19.

To install the Java Developers Kit

1 Download the Java 2 SDK from the following location:

`http://java.sun.com/products/archive/j2se/1.4.2_08/index.html`

2 Install the product in the default directory or in another directory of your choice.

Unpacking the ICM Software to a Staging Directory

Before installing third-party applications and configuring the Siebel ICM product, you download and install the base application to a staging directory. This is a user-defined temporary directory where you can install third-party software and customize the Siebel ICM files before deployment. In the procedures and examples that follow, this directory is identified as <STAGING>.

This task is a step in "Process of Meeting the Requirements for ICM Installation" on page 19.

To unpack the ICM software to a staging directory

1 On the Siebel ICM installation media, locate the Siebel_ICM-7.8.2.zip file.

2 Create a staging directory on your server machine, at a location and with a name of your choice.

NOTE: This location is referred to throughout this guide as the <STAGING> directory.

3 Extract the entire contents of the Siebel_ICM-7.8.2.zip file into the ICM <STAGING> directory.

Installing Mozilla Rhino

This procedure applies to all installations.

Mozilla Rhino is the Netscape Javascript Execution Engine. This component runs rule logic. It is required for ICM functionality.

This task is a step in "Process of Meeting the Requirements for ICM Installation" on page 19.

To install Mozilla Rhino

1 Download Mozilla Rhino from the following location:

`ftp://ftp.mozilla.org/pub/mozilla.org/js/rhino1_6R1.zip`

2 Unzip `rhino16R1.zip` to a temporary directory.

3 In the temporary directory, locate the js.jar file and copy it to the following location:

`<STAGING>/appserver/lib/third_party/`

Installing iText

This procedure applies to all installations.

iText generates PDF reports from Jasper report definitions. This component is required for ICM functionality.

This task is a step in "Process of Meeting the Requirements for ICM Installation" on page 19.

To install iText

1. Download iText from the following location:

   `http://prdownloads.sourceforge.net/itext/itext-1.01.jar?download`

2. Copy the itext-1.01.jar file to the following directory:

   `<STAGING>/appserver/lib/third_party/`

Installing jchardet

ICM requires all XML files to be encoded in UTF-8 format. jchardet detects the physical encoding of imported XML files. Without jchardet, if a user tries to import an XML file that is in another encoding, ICM imports and saves it as meaningless records, resulting in hard-to-delete random data. With jchardet, if a user tries to import an XML file that is in another encoding, the import fails and returns an error.

To install jchardet

1. Download jchardet from the following location:

   `http://sourceforge.net/project/showfiles.php?group_id=85452&release_id=171192`

2. Download the chardet.zip file.

3. From the ZIP file, extract chardet.jar and place it in the following directory:

   `<STAGING>/appserver/lib/third_party/`

Installing JDBC Drivers for Oracle

This procedure applies to Oracle installations only.

A JDBC driver allows Siebel ICM to connect to a data source to retrieve or update data. If you use an Oracle database, you must download an Oracle JDBC driver from Oracle's Web site.

This task is a step in "Process of Meeting the Requirements for ICM Installation" on page 19.

To install JDBC drivers for Oracle

1. Obtain the Oracle Driver driver for your version of Oracle.

   For the correct version number of the JDBC driver, see *System Requirements and Supported Platforms* on SupportWeb.

   NOTE: Siebel Systems does not provide the Oracle Driver file. You must acquire it yourself from your Oracle installation or through the Oracle TechNet Web site.

2 Place the Oracle Driver file in the following location:

`<STAGING>/db/lib/oracle/`

Do *not* unzip the Oracle Driver file.

Installing JDBC Drivers for DB2

This procedure applies to DB2 installations only.

A JDBC driver allows Siebel ICM to connect to a data source to retrieve or update data. IBM's DB2 fix pack includes JDBC driver jar files (for example, db2java.zip and db2jcc_license_cisuz.jar) used by some DB2 clients. After installing DB2, you can install those jar files.

This task is a step in "Process of Meeting the Requirements for ICM Installation" on page 19.

To install JDBC drivers for DB2

1 Navigate to <DB2_install_dir>/java/,and locate the following files:

`db2jcc.jar`
`db2jcc_license_cisuz.jar`

2 On the Siebel ICM installation media, locate the following file:

`db2jcc_license_cu.jar`

3 Copy the jar files that you located in Step 1 and Step 2 to <STAGING>/db/lib/db2/.

When you deploy Siebel ICM, the system copies the jar files to the application server directory.

Installing Apache Ant

This procedure applies to all installations.

Apache Ant allows you to run Ant target commands, which are used throughout the Siebel ICM installation process. For a listing of Ant targets and their definitions, see Appendix F, "Target Commands Reference."

This task is a step in "Process of Meeting the Requirements for ICM Installation" on page 19.

To install Apache Ant

1 Download apache-ant-1.5.4-bin.zip from the following location:

`http://archive.apache.org/dist/ant/binaries/`

2 Install the product in the default directory or in another directory of your choice.

NOTE: This install directory is referred to throughout this guide as <ANT>.

3 Navigate to <ANT>/bin and enter the following command:

```
Chmod 755 *
```

4 Add the <ANT>/bin directory to your $path environment variable.

Installing the X11 Server

This procedure applies to X11 installations only.

The X11 Server is an X virtual frame buffer server composed of two required packages, X11.vfb and X11.apps.clients.

To install the X11 Server

1 Verify that the two required packages, X11.vfb and X11.apps.clients, are installed by running the following commands:

```
lslpp -l X11.vfb
lslpp -l X11.apps clients
```

If they are installed, the Status field displays Committed.

2 If the required packages are not installed, run the following commands from directories containing their respective file sets to install them:

```
installp -gac -d. X11.vfb
installp -gac -d. X11.apps
```

3 Specify a display number.

Every X installation runs on its own display. By default, an X server runs on display :0. For the VFB Xserver, display :1 is recommended, unless another X server is running there.

- To set the display in a Borne-style shell (sh, bash, zsh, ksh), enter the following command:

  ```
  DISPLAY=:1 ; export DISPLAY
  ```

- To set the display in a C-style shell (csh, tcsh), enter the following command:

  ```
  setenv DISPLAY :1
  ```

4 To start the VFB X server, the following command is recommended:

```
/usr/bin/X11/X -force -vfb $DISPLAY & xhost +
```

This command starts the server and allows all clients to connect.

Installing hsqldb.jar for WebSphere

This procedure applies to WebSphere installations only.

The hsqldb.jar file is used for JMS Message persistence in WebSphere.

To install hsqldb.jar for WebSphere

1 Download the hsqldb.jar file from the following location:

`http://prdownloads.sourceforge.net/bulldog/hsqldb.jar?download`

2 Place the hsqldb.jar file in the following directory:

`<STAGING>appserver/lib/third_party/websphere5.1`

Downloading and Installing the Report Utility

This procedure applies only to installations that use the ICM Report Utility.

Jasper Reports is a tool for constructing and viewing reports within a Web application. Siebel ICM provides the Report Utility as a tool to help construct, test, and publish Jasper Reports. For complete instructions on installing and operating the Report Utility, see "Process of Installing the ICM Report Utility" on page 100.

Creating a Siebel ICM User

This procedure is optional for all installations.

It is recommended that you create a separate user to run Siebel ICM, following normal local user creation procedures. This guide assumes that you create such a user and name it `siebelicm`.

NOTE: If your application server is WebSphere, then create the siebelicm user with the sudo root privilege. The sudo privilege allows a non-root user to function as root user.

This task is a step in Processes for ICM Installation on page 15.

To create a Siebel ICM User

1 Create the siebelicm user. For information how to create users, consult your operating system's documentation.

2 Make sure this user has a home directory with sufficient space for the installation.

4 Siebel Incentive Compensation Management Installation

This chapter provides instructions for installing a stand-alone configuration of Siebel Incentive Compensation Management. All three tiers, including the application servers and databases, are hosted on the same machine. This chapter contains the following sections:

Process of Installing and Configuring ICM

This section summarizes the steps for installing and configuring Siebel Incentive Compensation Management, in the order in which they are performed.

NOTE: It is recommended that Siebel Incentive Compensation Management and its adjunctive applications be installed on a disk partition separate from the partition that hosts the operating system and system logs. It is not necessary to place tiers on individual partitions.

1 "Configuring the ICM Properties Files" on page 36

   a "Configuring ICM Deployment Properties" on page 37

   b "Configuring ICM Database Properties" on page 37

   c "Configuring ICM Application Server Properties" on page 39

   d "Configuring ICM Service Performance Properties" on page 40

   These tasks are required for all installations. You can perform these tasks in any order. The sequence shown here is recommended.

2 “Configuring Siebel CRM Integration Properties” on page 40

This task is required only if you are integrating ICM with Siebel CRM applications.

3 “Configuring Reports” on page 41

This task is required only if you are using the ICM Reports module.

4 “Configuring Number, Currency, Date, and Time Formats” on page 41

This task is optional.

5 “Preparing WebSphere for an ICM Deployment” on page 42

This task is required for WebSphere on UNIX only.

6 “Preparing to Integrate with Siebel CRM” on page 43

This task is required only if you are integrating ICM with Siebel CRM applications.

7 “Preparing for an ICM Deployment on WebSphere” on page 43

This task is required for WebSphere installations only.

8 “Deploying ICM on an Application Server” on page 44

This task is required for all installations.

9 “Starting the ICM Application Server” on page 46

- “Starting the JBoss Application Server” on page 46
- “Starting the WebSphere Application Server” on page 47

This task is required to verify installation. Follow the procedure appropriate to your application server.

10 “Running an ICM Instance for the First Time” on page 47

This task is required to verify installation and do initial setup.

11 “Stopping the ICM Application Server” on page 49

- “Stopping the JBoss Application Server” on page 49
- “Stopping the WebSphere Application Server” on page 50

Follow the procedure appropriate to your application server.

Configuring the ICM Properties Files

After you copy the ICM files into the <STAGING> directory, you need to locate the ICM properties files and enter certain settings. Some of these settings may vary depending on factors such as your application server, your database, and your company’s business practices.

The following topics describe configuration tasks:

- “Configuring ICM Deployment Properties” on page 37
- “Configuring ICM Database Properties” on page 37

- "Configuring ICM Application Server Properties" on page 39
- "Configuring ICM Service Performance Properties" on page 40
- "Configuring Siebel CRM Integration Properties" on page 40

Configuring the properties files in the order listed in this section is recommended.

Configuring ICM Deployment Properties

This procedure applies to all installations.

This section describes the procedure for configuring Siebel ICM's main deployment properties.

Within the deploy.default.properties properties file, lines beginning with deploy. are properties. Lines beginning with # are comments. The comments provide instructions on how to set the properties within the file.

This task is a step in "Process of Installing and Configuring ICM" on page 35.

To configure ICM deployment properties

1. Navigate to <STAGING>/deploy/.
2. Launch a text editor and open the deploy.default.properties file.
3. Read the comments for each property, and set the properties accordingly.

   The settings for critical properties that must be set are listed in the following table.

Property	Comments
deploy.app.name	Application name. Default is Siebel.
deploy.app.host	Machine names. Values are deploy.transactionDB.host, deploy.analyticsDB.host=localhost.
deploy.appserver	Name of the application server to which to deploy ICM. Values are websphere5.1 or JBoss3.0.
deploy.db	Database type.
deploy.logging.mode	Logging mode.

Configuring ICM Database Properties

This section describes the procedures for configuring Siebel ICM's database properties.

Follow the procedure that applies to the database for your installation.

Within the db.properties file, lines beginning with db. are properties. Lines beginning with # are comments. The comments provide instructions on how to set the properties within the file.

Configuring ICM for Oracle

You configure Siebel ICM for Oracle by performing the following procedure.

This task is a step in "Process of Installing and Configuring ICM" on page 35.

To configure ICM database properties for Oracle

1. Navigate to <STAGING>/deploy/oracle/.
2. Launch a text editor, and open the db.properties file.
3. Read the comments for each property, and set the property accordingly.

   NOTE: Make sure that the properties defined in the db.properties file match the values for your target Transaction and Analytics databases.

Configuring ICM for SQL Server

You configure Siebel ICM for SQL Server by performing the following procedure.

This task is a step in "Process of Installing and Configuring ICM" on page 35.

To configure ICM database properties for SQL Server

1. Navigate to <<STAGING>/deploy/sqlserver/.
2. Launch a text editor, and open the db.properties file.
3. Read the comments for each property, and set the property accordingly.

Configuring ICM for DB2 on UNIX

You configure Siebel ICM for DB2 by performing the following procedure.

This task is a step in "Process of Installing and Configuring ICM" on page 35.

To configure ICM database properties for DB2

1. Navigate to <STAGING>/deploy/db2/.
2. Launch a text editor, and open the db.properties file.
3. Read the comments for each property, and set the property accordingly.
4. Set the LOCKTIMEOUT parameter. See "Setting the LOCKTIMEOUT Parameter for DB2 on UNIX" on page 38.

Setting the LOCKTIMEOUT Parameter for DB2 on UNIX

In DB2, a query that reads a table can block a query that inserts or updates that table. The LOCKTIMEOUT parameter specifies how long the insert or update query waits for the read query to finish before closing and generating an error. For example, a LOCKTIMEOUT value of -1 closes the insert or update query immediately, while a value of 60 directs the query to wait 60 seconds.

You set the LOCKTIMEOUT parameter by performing the following procedure.

To set the LOCKTIMEOUT parameter

1. Connect to the ICM database by running the following command:

   ```
   db2 connect to <ICM_DB>
   ```

   For <ICM_DB>, substitute the name of your ICM database; for example, ICM314.

2. Find the LOCKTIMEOUT parameter by running the following command:

   ```
   db2 get database configuration
   ```

3. Set the LOCKTIMEOUT parameter by running the following command:

   ```
   db2 update db cfg using LOCKTIMEOUT 60
   ```

4. Make the changes effective by running the following command:

   ```
   db2 disconnect current
   ```

5. Reconnect to the ICM database.

6. Run the following command:

   ```
   db2 AUTOCONFIGURE using ISOLATION CS APPLY DB ONLY
   ```

   This sets the Transaction Isolation Level, a connection-specific property that defines how records are locked, to a system default of Cursor Stability (CS). This setting is suitable for systems like ICM that use numerous short-duration locks.

7. Disconect from the database by running the following command:

   ```
   db2 disconnect current
   ```

8. Restart the database manager by running the following command:

   ```
   db2 stop dbm followed by db2 start dbm
   ```

Configuring ICM Application Server Properties

Follow the procedure that applies to the application server for your installation.

This section describes the procedures for configuring Siebel ICM's application server properties.

Within the appserver.properties file, lines beginning with deploy. are properties. Lines beginning with # are comments. The comments provide instructions on how to set the properties within the file.

Configuring ICM for JBoss

You configure Siebel ICM for JBoss by performing the following procedure.

This task is a step in "Process of Installing and Configuring ICM" on page 35.

To configure ICM application server properties for JBoss

1 Navigate to <STAGING>/deploy/JBoss3.0/.

2 Launch a text editor, and open the appserver.properties file.

3 Read the comments for each property, and set the property accordingly.

Configuring ICM for WebSphere

You configure Siebel ICM for WebSphere by performing the following procedure.

This task is a step in "Process of Installing and Configuring ICM" on page 35.

To configure ICM application server properties for WebSphere

1 Navigate to <STAGING>/deploy/websphere5.1/.

2 Launch a text editor, and open the appserver.properties file.

3 Read the comments for each property, and set the property accordingly.

Configuring ICM Service Performance Properties

This procedure applies to all installations.

This section describes the procedure for configuring Siebel ICM's service performance properties.

Within the service.properties file, lines beginning with service. are properties. Lines beginning with # are comments. The comments provide instructions on how to set the properties within the file.

This task is a step in "Process of Installing and Configuring ICM" on page 35.

To configure ICM service performance properties

1 Navigate to <STAGING>/deploy/services/.

2 Launch a text editor, and open the service.properties file.

3 Read the comments for each property, and set the property accordingly.

Configuring Siebel CRM Integration Properties

This task is required only if you are integrating ICM with Siebel CRM applications.

This section describes the procedure for configuring Siebel ICM-to-Siebel CRM integration properties.

Within the siebel.crm.properties file, lines beginning with siebel. are properties. Lines beginning with # are comments. The comments provide instructions on how to set the properties within the file.

This task is a step in "Process of Installing and Configuring ICM" on page 35.

To configure Siebel CRM integration properties

1. Navigate to <STAGING>/deploy/siebel_crm_interface.
2. Launch a text editor, and open the siebel.crm.properties file.
3. Read the comments for each property, and set the property accordingly.

Configuring Reports

This procedure applies only to installations that use the ICM Report Utility.

The ICM Reports Utility allows you to run ICM reports, either as a stand-alone application or integrated with Siebel ICM. For details on configuring the ICM Report Utility, see "Process of Configuring the ICM Report Utility" on page 102.

Configuring Number, Currency, Date, and Time Formats

ICM supports multiple locales simultaneously. The system can have users in languages such as English, French, and Portuguese, all on the system at the same time. At login time, each user selects his or her preferred locale.

The uiFormatConfig.xml file contains ICM's instructions for displaying number, currency, date, and time formats for each locale. This file is populated with the standard defaults for all the locales that ICM supports. Optionally, you can customize the number, currency, date, or time formats for one or more of these locales.

For example, suppose your company wants to display American English locale dates in the format 01-23-2005 instead of in the locale's default of 01/23/2005. In this case, you could change the date format for the American English locale only.

You can perform a similar customization for number formats. By default, ICM shows two degrees of decimal precision; for example, 1.23. If you want, you can configure ICM to display five degrees of precision; for example, 1.23563. Again, this is done by locale only.

You can customize a locale's formats for the entire ICM instance, or for each OU. For example, you can specify one date format for the French locale for OU1, and another date format for the French locale for OU2.

For more information on Java number formats, see the following Web page:

`http://java.sun.com/j2se/1.4.2/docs/api/java/text/DecimalFormat.html`

For more information on Java date formats, see the following Web page:

`http://java.sun.com/j2se/1.4.2/docs/api/java/text/SimpleDateFormat.html`

To configure number, currency, date, and time formats

1. Navigate to <STAGING>/config/.

2 Launch a text editor, and open the uiFormatConfig.xml file.

3 Read the instructions in the file and perform your configuration changes accordingly.

4 Save the file, and close it.

NOTE: For the settings in the uiFormatConfig.xml file to take effect, you must start (or restart) the application server.

Preparing WebSphere for an ICM Deployment

This procedure applies to WebSphere installations only.

To prepare WebSphere on UNIX for an ICM deployment, you remove the previous ICM installation, if any, and the WebSphere Sample Applications.

If you are redeploying Siebel ICM to WebSphere on UNIX, you must remove any previous installation of the same name before running a deploy, deploy-appserver, or fast-deploy target. Also, WebSphere includes several sample applications. Remove these sample applications so that Siebel ICM can operate in secured mode.

To remove applications from WebSphere

1 If your WebSphere server is not running already, start it up.

2 Start the WebSphere Administrative Console.

3 Open a browser window and go to the following URL:

`http://<hostname>:<port>/admin`

Where <hostname> is the full network name of the appserver machine and <port> is the Administrative Console port, usually 9090. This port is specified by deploy.port.admin.HttpConnector in appserver.properties.

4 Log in as the Siebel ICM system administrator.

Default user name is mark and default password is anil.

5 Navigate to Applications/Enterprise Applications.

6 Perform the following steps:

- To remove a previous ICM installation, select the Siebel ICM application (default name is Siebel), and click Uninstall.
- To remove the WebSphere sample applications, select each sample application, and click Uninstall.

7 To commit the change to the WebSphere configuration storage, click the Save tab.

Preparing to Integrate with Siebel CRM

This task is required only if you are integrating ICM with Siebel CRM applications.

You can configure Siebel ICM to extract data from Siebel CRM applications and authenticate users against Siebel CRM. For information on configuring ICM for user authentication against Siebel CRM, see the authentication topics in Chapter 5, "Postinstallation Tasks."

You must first install and configure Siebel CRM. Then you must copy certain JAR files shipped with Siebel CRM into the ICM files. For convenience, copies of these JAR files come with Siebel ICM. However, it is highly recommended that you copy the JAR files from Siebel CRM into ICM. This precaution prevents version mismatches between ICM and Siebel CRM.

This task is a step in "Process of Installing and Configuring ICM" on page 35.

To copy integration JAR files from Siebel CRM into ICM

1. On the machine where Siebel CRM is installed, navigate to the following directory:

   `siebsrvr/CLASSES`

2. Locate the Siebel.jar file and copy it to a temporary directory.

3. Rename the copy as SiebelJDB.jar.

   CAUTION: Do *not* rename the file in the CLASSES directory.

4. Locate the JAR files in the CLASSES directory that are named SiebelJI_xxx.jar, where xxx is a Siebel CRM language code like ENU or PSL. These are the language pack JAR files. Examples:

   ```
   SiebelJI_enu.jar
   SiebelJI_psl.jar
   ```

5. Copy SiebelJDB.jar and the language pack JAR files to the following directory:

   `<STAGING>/appserver/lib/siebel_crm/7.8`

   The three files in this directory (SiebelJDB.jar, SiebelJI_enu.jar, and SiebelJI_psl.jar) must be the same as those of the Siebel CRM installation. The system deploys these Siebel CRM JAR files along with the ICM application files when you complete the task described in "Deploying ICM on an Application Server" on page 44.

Preparing for an ICM Deployment on WebSphere

This procedure applies to WebSphere installations only.

Before deploying ICM on WebSphere, you must prepare the system as described in this section.

This task is a step in "Process of Installing and Configuring ICM" on page 35.

To prepare to deploy ICM on a WebSphere application server

1 If you are deploying ICM on WebSphere, log on as the root user.

Because the installation script modifies WebSphere's configuration file, the user account to deploy Siebel ICM must have write permission to WebSphere's files. It is recommended that you use the same user account that installed WebSphere to deploy Siebel ICM; in other words, the root user.

2 In a command window, run the following command:

```
<DEPLOYMENT>/Websphere/appserver/bin/setupcmdline.sh
```

This command sets up various environment variables correctly.

NOTE: Step 2 through Step 4 should be run in the same window.

3 Check the version of Java that is first in your path by typing the following:

```
java -version
```

This command should return the Java version that your ICM installation uses. For the specific version, see *System Requirements and Supported Platforms*.

This command should *not* return 1.3.1. If it does, make sure `Oracle/java/bin` is not first in your path. Ideally the IBM Java should be first in your path. You can make the IBM Java first in your path by setting it as follows:

```
PATH=%WAS_PATH%
```

4 In appserver.properties, the properties `deploy.cell.name` and `deploy.node.name` are case sensitive. Their values MUST EXACTLY MATCH the names given during WebSphere installation. Otherwise you receive the following error message:

```
ADMA5026E: No valid target is specified in ObjectName {0} for module {1}.
```

This error means that the target server or cluster specified for a module in application installation does not exist. The target server name is specified as WebSphere:cell=cellName,node=nodeName,server=serverName. The target cluster name is specified as WebSphere:cell=cellName,cluster=clusterName.

Deploying ICM on an Application Server

This procedure applies to all installations.

After configuring ICM for installation by altering files in the <STAGING> directory, you deploy the application. The <DEPLOYMENT> directory is the application server root directory to which Siebel ICM is deployed. The deployment step overwrites some of the files in the <DEPLOYMENT> directory with customized files from the <STAGING> directory.

This task is a step in "Process of Installing and Configuring ICM" on page 35.

To deploy ICM on an application server

1 Open a command window and navigate to the <STAGING>/deploy/ directory.

2 Make sure that the JAVA_HOME variable points to the installation location of your JVM by performing the following steps:

   a Add JAVA_HOME to the environment variables in your ksh or bash profile.

   b Perform one of the following steps:

      ❑ For WebSphere, set the JAVA_HOME property to the following path:

      <DEPLOYMENT>/WebSphere/Appserver/java

      ❑ For JBoss, set the JAVA_HOME property to the location where you installed the JDK, referred to as the <JAVA_HOME> directory.

3 If you are installing DB2 on AIX, you must set the environment variables that follow.

   For ksh and bash profiles, add the following to the .profile file:

```
DB2HOME=/instances/v8inst1
LIBPATH=$LIBPATH:$DB2HOME/sqllib/lib
Export LIBPATH
if [ -f $DB2HOME/sqllib/db2profile ]; then
.  $DB2HOME/sqllib/db2profile
fi
```

   Run any further ant targets from this shell.

4 Deploy the ICM application. At the command prompt, perform one of the following steps:

   ■ To rebuild everything including the database, run the following target:

   ant deploy

   ■ To rebuild everything *except* the database, run the following target:

   ant fast-deploy

   CAUTION: If you are upgrading from an earlier version of ICM to the current version, do *not* use ant deploy. Instead, use ant fast-deploy. For information about upgrades, see Appendix A, "Upgrading from ICM 7.8 to 7.8.2."

5 If you are installing DB2, to help improve performance, it is recommended that you make all tables volatile by running the enable_volatile_tables.sql script.

   a Navigate to <STAGING>/db/sql/common/db2.

   b Open the enable_volatile_tables.sql file and decide whether it is appropriate for your system.

   c Connect to the Siebel ICM transaction database with the transaction database user that Siebel ICM is set to use.

   d Run the following command:

```
db2 –td@ -f enable_volatile_tables.sql
```

CAUTION: If you need to change a configuration after it has been installed, do not modify or delete files in the <DEPLOYMENT> directory. Instead, change the source files in the <STAGING> directory and run an ant target again to deploy the change. You must use this method because edits made directly to the <DEPLOYMENT> directory are overwritten when ant targets copy files from the <STAGING> directory. For information on how these targets operate, see Appendix F, "Target Commands Reference."

Starting the ICM Application Server

After you have installed the ICM software, you can start the application server. This section describes the following procedures:

- "Starting the JBoss Application Server" on page 46
- "Starting the WebSphere Application Server" on page 47

Starting the JBoss Application Server

This procedure applies to JBoss installations only.

This section describes the procedure for starting Siebel ICM's JBoss application server.

This task is a step in "Process of Installing and Configuring ICM" on page 35.

To start the JBoss application server

1. Make sure the transaction database and the analytics database are running.
2. Open a command window, and change the directory to the following path:

   `<DEPLOYMENT>/bin`
3. To start the Siebel ICM server, at the command prompt, enter `run.sh`.

   NOTE: If you have installed JBoss as a service, you must use the Windows Service menu to start JBoss.

   The Command window displays startup messages and errors. Siebel ICM archives important messages in the following location:

   `<DEPLOYMENT>/server/default/log`

   When the server has started, the message Started Successfully at <time> appears in the command window.

   CAUTION: You can minimize the command window, but *do not* close it. Closing the command window shuts down the server. Logging out of the current session also shuts down the server.

Starting the WebSphere Application Server

This procedure applies to WebSphere installations only.

This section describes the procedure for starting Siebel ICM's WebSphere application server.

NOTE: On AIX, IBM recommends starting the server as the root user.

This task is a step in "Process of Installing and Configuring ICM" on page 35.

To start the WebSphere application server

1 Open a command window, and change the directory to the following:

```
<DEPLOYMENT>/WebSphere/AppServer/bin
```

2 Start the server by entering the following command:

```
sh startServer.sh server1 -username <USERNAME> -password <PASSWORD>
```

For <USERNAME>, substitute your ICM administrator user name; default is `mark`. For <PASSWORD>, substitute your ICM administrator password; default is `anil`.

NOTE: You can change this default ICM user name and password through WebSphere. For information, see Changing the ICM Username and Password for WebSphere on page 51.

Running an ICM Instance for the First Time

This procedure applies to all installations.

After you have started the ICM application server, you can launch ICM and begin to configure the environment. Configuring the environment is explained in *Siebel Incentive Compensation Management Configuration Guide*.

This task is a step in "Process of Installing and Configuring ICM" on page 35.

Launching ICM for the First Time

Use this procedure to launch your ICM installation for the first time.

To launch ICM for the first time

1 If you have not done so already, start the application server. See "Starting the ICM Application Server" on page 46.

2 At the command prompt, navigate to <STAGING>/deploy and populate the dictionary and the system by running the following target:

```
ant populate-all
```

3 Continue populating the dictionary and system by performing either of the following procedures:

- To add your company's data to this ICM instance, follow the steps of "Creating Your Own Data in ICM" on page 48.
- (Optional) To add fictitious sample data to this ICM instance, follow the steps of "Setting Up Sample Data in ICM" on page 48.

Creating Your Own Data in ICM

Use this procedure to create your own data in ICM.

To create your own data in ICM

1. Launch an Internet browser and enter the appropriate address.
   - For JBoss installations:

     `http://<hostname>:8080/Siebel/bootstrap`

     NOTE: 8080 is the default http port for Tomcat. Your address depends on the setting of the deploy.port.tomcat.HttpConnector property in the appserver.properties property file.
   - For WebSphere installations:

     `http://<hostname>:9080/Siebel/bootstrap`

     NOTE: 9080 is the default http port for WebSphere. Your address depends on the setting of the deploy.port.websphere.HttpConnector property in the appserver.properties property file.
2. On the Siebel ICM login screen, log in with the default system administrator user name (mark) and password (anil) that are shipped with the application.
3. Create an enterprise unit and an operating unit in the Bootstrap URL.

   Creating these units allows you to get in to the system. For information on how to create enterprise units and operating units, see the chapter on setting up access to ICM in *Siebel Incentive Compensation Management Configuration Guide*.
4. Launch the Siebel ICM UI in an Internet browser by entering the appropriate address.
   - For JBoss installations:

     `http://<hostname>:8080/Siebel/logon.jsp`
   - For WebSphere installations:

     `http://<hostname>:9080/Siebel/logon.jsp`
5. On the Siebel ICM login screen, log in with the operating unit user name and password you created through the bootstrap URL.

Setting Up Sample Data in ICM

Use this procedure to set up sample data in ICM.

To set up sample data in ICM

1 At the command prompt, to populate the ICM deployment with sample data, perform either of the following steps:

- For the Blink dataset, run the following target:

  ```
  ant populate-blink
  ```

- For the Daytona dataset, run the following target:

  ```
  ant populate-daytona
  ```

2 Launch the Siebel ICM UI in an Internet browser by entering the appropriate address.

- For JBoss installations:

  ```
  http://<hostname>:8080/Siebel/logon.jsp
  ```

- For WebSphere installations:

  ```
  http://<hostname>:9080/Siebel/logon.jsp
  ```

3 On the Siebel ICM login screen, log in to the sample data.

Administrator usernames and passwords are listed in the following table.

Dataset	Unit	Username	Password
Blink	Enterprise Unit	blinkeu1	demo1234
Blink	Operating Unit	blinkou1	demo1234
Daytona	Enterprise Unit	DaytonaEU	ce1234
Daytona	Operating Unit	Daytona	ce1234

Stopping the ICM Application Server

After you have completed the initial ICM configuration and closed the application, you can stop the ICM application server. This section describes the following procedures:

- “Stopping the JBoss Application Server” on page 49
- “Stopping the WebSphere Application Server” on page 50

Stopping the JBoss Application Server

This procedure applies to JBoss installations only.

This section describes the procedure for stopping Siebel ICM’s JBoss application server.

This task is a step in “Process of Installing and Configuring ICM” on page 35.

To stop the JBoss application server

1. Log out of the application UI.
2. Open a command window, and change the directory to the following path:

   `<DEPLOYMENT>/bin`

3. To stop Siebel ICM, at the command prompt, enter `shutdown.sh`.

   NOTE: If you have installed JBoss as a service, you must use the Windows Service menu to stop JBoss.

   When the services have completely shut down, the Siebel ICM service window closes.

Stopping the WebSphere Application Server

This procedure applies to WebSphere installations only.

This section describes the procedure for stopping Siebel ICM's WebSphere application server.

NOTE: On AIX, IBM recommends stopping the server as the root user.

This task is a step in "Process of Installing and Configuring ICM" on page 35.

To stop the WebSphere application server

1. Open a command window, and change the directory to the following path:

   `<DEPLOYMENT>/AppServer/bin`

2. Stop the server by entering the following command:

   `sh stopServer.sh server1 -username <USERNAME> -password <PASSWORD>`

   For `<USERNAME>`, substitute your ICM administrator user name; default is `mark`. For `<PASSWORD>`, substitute your ICM administrator password; default is `anil`.

5 Postinstallation Tasks

This appendix describes how to make various post-installation changes to ICM. Some of these tasks apply to WebSphere installations only; others can apply to any system. This appendix contains the following sections:

Changing the ICM Username and Password for WebSphere

You can create a new administrator account by entering a new username and password.

To change the ICM username and password for WebSphere

1. Start WebSphere with the old username and password.
2. Log on to the WebSphere Administrative Console with the old username and password.
3. Create a J2C authentication alias from the Administrative Console by performing the following steps:
   a. Navigate to Security > JAAS Configuration > J2C Authentication Data page and click New.
   b. In the J2C Configuration dialog box, enter a new username in the Alias and Username fields and a new password in the Password field.
   c. Save the changes.
4. Update the security by performing the following steps:
   a. Navigate to Security > User Registries > Custom page.
   b. Change User Id to the new username and Password to the new password.
   c. Save the changes.

5 Create a new administrator account by performing the following steps:

   a Navigate to Resources > Websphere JMS Provider> Websphere Topic Connection Factories > topicConnectionFactory page.

   b Change the Component Managed Authentication Alias and Container Managed Authentication Alias drop-down list to <CELL_NAME>/new_username alias.

   c Save the change.

6 Navigate to the <STAGING>/deploy directory, and generate the encrypted username and password by running the following ant targets:

   ```
   ant encrypt-text -Dtext=<username>
   ```

   For <username>, substitute the plain text username to be encrypted.

   ```
   ant encrypt-text -Dtext=<password>
   ```

   For <password>, substitute the plain text password to be encrypted.

7 Navigate to <DEPLOYMENT>/Siebel/conf/node/server1/, open the SiebelICMConfig.xml file, and change the system.admin.username and system.admin.password properties to the values in Step 6.

8 Restart WebSphere with the new username and password.

Changing the ICM Password Only for WebSphere

You can modify the default account by changing the password only.

To change the ICM password only for WebSphere

1 Start WebSphere with the old username and password.

2 Log on to the WebSphere Administrative Console with the old username and password.

3 To change only the password, perform the following steps:

   a Navigate to Security > JAAS Configuration > J2C Authentication Data page and click <CELL_NAME>/mark alias.

   b In the J2C Configuration dialog box, enter a new password in the Password field.

   c Save the change.

4 Update the security by performing the following steps:

   a Navigate to Security > User Registries > Custom page.

   b Change Password to the new password.

   c Save the change.

5 Navigate to the <STAGING>/deploy directory, and generate the encrypted username and password by running the following ant target:

```
ant encrypt-text -Dtext=<password>
```

where <password> is the plain text password to be encrypted.

6 In <DEPLOYMENT>/Siebel/conf/node/server1/SiebelICMConfig.xml, change the system.admin.password property to the value in Step 6.

7 Restart WebSphere with the old username and the new password.

Deploying the Precompiled JSPs for WebSphere

If you have deployed Siebel ICM but have not set the deploy.jsp.precompilation property in the websphere5.1/appserver.properties file, you can still deploy the precompiled JSPs only.

To deploy the precompiled JSPs for WebSphere

1 Open the <STAGING>/deploy/websphere5.1/appserver.properties file.

2 Change the deploy.jsp.precompilation property value to true.

3 Navigate to <STAGING>/deploy.

4 Run the following target:

```
ant fast-deploy
```

Changing the Application Session Timeout for WebSphere

The session timeout value indicates the amount of time that an end user can be inactive before the user session times out. By default, the session timeout is 30 minutes.

Two levels of control apply to the session timeout. One level is in the server Web container, and the other level is in the application. Application-level control overrides Web container-level control. Use this procedure to change the application-level session timeout value.

To change the application session timeout for WebSphere

1 Log on to the WebSphere Administrative Console.

2 On the menu on the left side of the screen, navigate to Applications > Enterprise Applications.

3 In the main window, select the Siebel application.

4 Under Additional Properties, select the Session Management link.

5 Adjust Session Timeout property to the desired value, and click Apply.

6 On the menu bar near the top left, click Save.

NOTE: You may need to stop and restart WebSphere for this change to take effect.

Populating the Update Analytics Stored Procedures

After restoring a database from one system to another (for example, from a production environment to a test environment), and before starting the application server, you must populate the Update Analytics stored procedures in the databases with the correct server information.

To populate the Update Analytics stored procedures

1. Restore the database to the target system.
2. Navigate to the <DEPLOYMENT> directory on the machine on which your application server is installed.
3. At the command line, enter the following command:

   ```
   ant db-finish
   ```

   This command populates the Update Analytics stored procedures in the databases with the correct server information.

   CAUTION: If this step is not performed, the Update Analytics process fails.
4. Start the application server.

Configuring Apache as HTTP Server for JBoss

Siebel ICM on JBoss uses Tomcat as the default HTTP Server. You can use Apache as the HTTP Server in place of Tomcat. Tomcat will still generate the dynamic pages.

For more information about Apache, see the Jakarta Apache documentation.

To configure Apache as HTTP server for JBoss

1. Navigate to the Web page http://httpd.apache.org/ and do the following:
   a. Download the httpd-2.0.54.tar.gz file.
   b. Extract the httpd-2.0.54.tar.gz file and use it to compile and install Apache.

      For detailed instructions, see the Apache documentation.
2. Navigate to the Web page http://www.reverse.net/pub/apache/jakarta/tomcat-connectors/jk2/binaries/solaris/ and do the following:
   a. Download the jakarta-tomcat-connectors-jk2.0.2-solaris8-apache2.0.43.tar.gz file.
   b. From the jakarta-tomcat-connectors-jk2.0.2-solaris8-apache2.0.43.tar.gz file, extract the mod_jk2.so file and place it in the <APACHE HOME>\modules directory.
3. Navigate to the <APACHE HOME>\conf\ directory and do the following:

a Open the httpd.conf file in a text editor.

b Locate the LoadModule (Dynamic Shared Object (DSO) Support) section and add the following line:

```
LoadModule jk2_module modules/mod_jk2.so
```

4 In the <APACHE HOME>\conf\ directory, create a file named workers2.properties.

This file sets up the workers that carry out the work of the servlet container.

5 In the workers2.properties file, add lines to specify the necessary directives.

The following directives are needed for the JK2 connector to function:

- The SHM (shared memory) file directive. This entry must match the shared memory directive in the jk2.properties file (see Step 6).
- The worker directive. The worker is a TCP socket connection named channel.socket.
- A URI directive. The URI translates your requests to Tomcat.

For example:

```
#define the shared memory file
[shm]
file=/Apache2/logs/jk2.shm

# Define the communication channel
[channel.socket:localhost:8009]
tomcatId=localhost:8009
[ajp13:localhost:8009]
channel=channel.socket:localhost:8009

#define the URI directive
[uri:/Siebel/*]
worker=ajp13:localhost:8009
```

6 Navigate to the <DEPLOYMENT>\jboss-3.0.3_tomcat-4.1.12\tomcat-4.1.x\conf directory and open the jk2.properties file in a text editor.

This file contains the JK2 configuration information.

7 Modify the jk2.properties file as needed to set up your shared memory (SHM file) directive.

This directive is necessary for Apache and Tomcat to communicate. The default is as follows:

```
#Shared memory directive
shm.file=/Apache2/logs/jk2.shm
```

The <APACHE_HOME>/logs directory is the recommended location for the SHM file. Although you can name the file anything, the recommended file name is jk2.shm.

8 In the <DEPLOYMENT>\jboss-3.0.3_tomcat-4.1.12\tomcat-4.1.x\conf directory, open the server.xml file in a text editor and uncomment the following lines:

```
<Connector className="org.apache.coyote.tomcat4.CoyoteConnector"
port="8009" minProcessors="5" maxProcessors="75"
enableLookups="true" redirectPort="8443"
```

```
acceptCount="10" debug="0" connectionTimeout="20000"
useURIValidationHack="false"
/>
```

This sets the connector adaptor to listen on Port 8009, the default port for the JK2 connector.

9 To execute the configuration changes, do the following, in the order shown:

a Restart Tomcat.

b Restart the Apache server.

Configuring Environment Settings for LDAP Authentication

LDAP Authentication is an optional authentication mode. You can configure the LDAP Authentication after Siebel ICM is installed. The LDAPLoginModule is a LoginModule implementation that authenticates against an LDAP server using JNDI login with the login module configuration options. You use the LDAPLoginModule if your username and credential information are stored in an LDAP server that is accessible using a JNDI LDAP provider.

The LDAP connectivity information is provided as configuration options that are passed through to the environment object used to create JNDI initial context.

To configure your environment settings for LDAP authentication, you need to set the property options in the login configuration file, which is different for each application server, to values that make sense for your environment, as described in the following procedure. The environment settings are located in the application launcher of the operating system you are using. Environment settings differ depending on which OS you use.

NOTE: The following procedure is optional. You do not need configure your environment settings unless you have a different set of environment settings.

After configuring the environmental settings, if needed, you must perform one of the following procedures, as appropriate to your application server platform:

- “Configuring LDAP Authentication for JBoss” on page 61
- “Configuring LDAP Authentication for WebSphere” on page 64

To configure LDAP environment settings

1 Set the appropriate LDAP JNDI properties by using the recommended options in the following table.

Properties	Comments
java.naming.factory.initial	The class name of the InitialContextFactory implementation.
java.naming.provider.url	The LDAP URL for the LDAP server.
java.naming.security.authentication	The security level to use.

Properties	Comments
java.naming.factory.initial	The class name of the InitialContextFactory implementation.
java.naming.provider.url	The LDAP URL for the LDAP server.

2 Set the properties options by using the recommended options in the following table.

Property	Comments
matchOnUserDN=true|false	A flag indicating if the search for user roles should match on the user's fully distinguished name. If false, just the username is used as the match value against the uidAttributeName attribute. If true, the full userDN is used as the match value.
unauthenticatedIdentity=string	The principal name that should be assigned to requests that contain no authentication information. This behavior is inherited from the UsernamePasswordLoginModule superclass.
password-stacking=useFirstPass	When the password-stacking option is set, this module first looks for a shared username and password under the property names javax.security.auth.login.name and javax.security.auth.login.password in the login module shared state map. If these elements found, they are used as the principal name and password. If these elements are not found, the principal name and password are set by this login module and stored under the property names javax.security.auth.login.name and javax.security.auth.login.password.
allowEmptyPasswords	A flag indicating if empty (length 0) passwords should be passed to the LDAP server. An empty password is treated as an anonymous login by some LDAP servers, and this may not be a desirable feature. Set this to false to reject empty passwords, true to have the LDAP server validate the empty password. The default is true. For more information about the configuration options, see "Common Valid Options to Set for the LDAP Login Module" on page 58.

Common Valid Options to Set for the LDAP Login Module

The authentication of a user is performed by connecting to the LDAP server based on the login module configuration options. Table 5 lists some of those options. Connecting to the LDAP server is done by creating an InitialLdapContext with an environment composed of the LDAP JNDI properties described in "Configuring Environment Settings for LDAP Authentication" on page 56. The Context.SECURITY_PRINCIPAL is set to the distinguished name of the user as obtained by the callback handler in combination with the principalDNPrefix and principalDNSuffix option values. The Context.SECURITY_CREDENTIALS property is set either to the String password or to the Object credential, depending on the useObjectCredential option.

Table 5. Common Valid Options to Set for the LDAP Login Module

Option Name	Description	Default
LdapContextFactory	The class name of the context factory of the LDAP provider you are using.	`com.sun.jndi.ldap.LdapCtxFactory`
authentication-type	Types include: ■ none. Access to LDAP does not require the name or password of the principal user. ■ simple. Access to LDAP requires a user name and password. These values are sent to LDAP in plain text. ■ CRAM-MD5. Access to LDAP requires a user name and password. These values are sent to LDAP in encrypted text.	`simple`
protocol	If the protocol is SSL, then the port is 636.	
host	The machine name of the LDAP server.	`icmwin2k01.corp.siebel.com`
port	The port the LDAP Server is running on.	`15774`

Table 5. Common Valid Options to Set for the LDAP Login Module

Option Name	Description	Default
DNPrefix	A prefix to add to the username when forming the user distinguished name. This is useful if you prompt a user for a username and you do not want them to have to enter the fully distinguished name. Using this property and DNSuffix the userDN is formed as follows: `String userDN = principalDNPrefix + username + principalDNSuffix;`	`uid=`
DNSuffix	A suffix to add to the username when forming the user distinguished name. This is useful if you prompt a user for a username and you do not want them to have to enter the fully distinguished name. Using this property and DNPrefix the userDN is formed as follows: `String userDN = principalDNPrefix + username + principalDNSuffix;`	`,ou=People,dc=corp,dc=siebel,dc=com`

Additionally, The Login Module options include whatever options your LDAP JNDI provider supports. Examples of standard property names are:

`"Context.INITIAL_CONTEXT_FACTORY = "java.naming.factory.initial"`

`"Context.SECURITY_PROTOCOL = "java.naming.security.protocol"`

`"Context.PROVIDER_URL = "java.naming.provider.url"`

`"Context.SECURITY_AUTHENTICATION = "java.naming.security.authentication"`

Configuring ICM Authentication Modes for JBoss

This section explains how to configure authentication modes for JBoss.

NOTE: Siebel ICM includes built-in authentication with user names and passwords stored in the Siebel ICM database. (In the login-config.xml file, this is referenced as Castor authentication.) For this mode, no configuration is necessary. This section is applicable *only* if your installation uses a mode other than the built-in Siebel ICM authentication.

The process of configuring authentication modes for JBoss consists of the following procedures:

- "Configuring Alternate Authentication Modes for JBoss" on page 60

Configuring Alternate Authentication Modes for JBoss

To configure an alternate (that is, other than the ICM default) authentication mode for ICM on JBoss, follow this procedure:

To configure an alternate authentication mode for JBoss

1. Navigate to the following directory:

   `<STAGING>\etc\JBoss3.0\server\default\conf`

2. In a text editor, open the login-config.xml.in file.

   The login-config.xml.in file contains several login modules. The login modules are chained. If supplied credentials fail authentication against the first login module, the system moves on to the next, and so on.

   ICM ships with only the CastorLoginModule configured. This login module authenticates administrator and participant user names and passwords against the ICM database.

3. To use a login module other than the CastorLoginModule, comment out the CastorLoginModule with authenticate-administrators and authenticate-participants set to true.

4. Do one of the following:

   - If you will be authenticating administrators against an alternate authentication mode, go to Step 5.
   - If you still want to authenticate administrators against the CastorLoginModule (most companies use this type of authentication), uncomment the CastorLoginModule with authenticate-administrators set to true and authenticate-participants set to false.

5. Depending on the authentication mode you want to configure, continue to one of the following procedures:

   - "Configuring Siebel Databean Authentication for JBoss" on page 60
   - "Configuring LDAP Authentication for JBoss" on page 61

Configuring Siebel Databean Authentication for JBoss

This procedure applies only to installations authenticating users against the Siebel Enterprise Server.

This authentication mode allows users to be authenticated against an instance of Siebel Enterprise Server. The options listed in this section are used to build the connect string needed by the Siebel Data Bean login method.

To configure Siebel Databean Authentication for JBoss

1. Follow the steps of "Configuring Alternate Authentication Modes for JBoss" on page 60.
2. Navigate to <STAGING>/deploy/siebel_crm_interface.
3. Open the siebel.crm.properties file in a text editor.
4. Set the connect string and other properties to values for connecting to your Siebel Server.

   For instructions on how to set these properties, see the section about using the EAI HTTP Transport for inbound integration in *Transports and Interfaces: Siebel Enterprise Application Integration*.
5. Navigate to <STAGING>/etc/JBoss3.0/server/default/conf/
6. Open the login-config.xml.in file in a text editor.
7. In the login-config.xml.in file, uncomment the following section:

```
<login-module code = "com.motiva.ce.security.spi.SiebelLoginModule" flag =
"sufficient">
  <module-option name = "unauthenticatedIdentity">nobody</module-option>
  <module-option name = "multi-threaded">true</module-option>

  <module-option name="connectString">@siebel.connection.string@</module-option>
  <module-option name="language">@siebel.language@</module-option>
</login-module>
```

   siebel.connection.string and siebel.language are replaced during the deployment build with values taken from the siebel.crm.properties file.

Configuring LDAP Authentication for JBoss

You can configure Siebel ICM so that it authenticates users against an LDAP server. To do this for JBoss, after you have verified your operating system's environment settings in "Configuring Environment Settings for LDAP Authentication" on page 56, you must configure the JBoss Login Module.

To configure LDAP authentication for JBoss

1. Follow the steps of "Configuring Alternate Authentication Modes for JBoss" on page 60.
2. Uncomment or cut and paste the LDAP login module definition, and fill in the appropriate values in the module-option XML elements as shown in the following code example:

```
<!--
ALTERNATE AUTHENTICATION - LDAP -->
<login-module code="com.motiva.ce.security.spi.LDAPLoginModule" flag="sufficient">
<module-option name="unauthenticatedIdentity">nobody</module-option>
<module-option name="authentication-type">simple</module-option>
<module-option name="protocal"></module-option>
<module-option name="host">icmwin2k01.corp.siebel.com</module-option>
<module-option name="port">15774</module-option>
```

```
<module-option name="DNPrefix">uid=</module-option>
<module-option name="DNSuffix">,ou=People,dc=corp,dc=siebel,dc=com</module-option>
</login-module>
```

3 Navigate to the <STAGING>\deploy directory and run the following target:

```
ant deploy-etc
```

4 Restart JBoss.

Configuring ICM Authentication Modes for WebSphere

ICM supports login module chain functionality. By default, the chain includes AdminLoginModule and CastorLoginModule. You can implement a login module following JAAS SPEC and plug it into the chain. ICM provides the following login modules:

- CastorLoginModule (default)
- SiebelLoginModule
- LDAPLoginModule

ICM can chain AdminLoginModule with one of the provided login modules or with any single customer-implemented Login Module. You can replace CastorLoginModule with one of the other login modules.

This section explains how to configure authentication modes for WebSphere. This section includes the following topics:

Confirming the CastorLoginModule Setting

CastorLoginModule is the default Login Module that comes with ICM. You can check this setting by completing the following procedure.

To confirm the CastorLoginModule setting

1 In WebSphere, navigate to Security > JAAS Configuration > Application Logins.

2 Click CE8-For-CEUserRegistry.

3 Click JAAS Login Modules.

4 Click the first module, and then click Custom Properties.

5 Click the delegate property.

6 Change the value of the delegate property to com.motiva.ce.security.spi.CastorLoginModule if it has any other value.

Configuring Alternate Authentication Modes for WebSphere

For any non-default login module to work, the CastorLoginModule must not authenticate participants. In almost all cases, where administrator accounts exist only in ICM, the CastorLoginModule continues to authenticate administrators. Only in rare situations, where both participant and administrator accounts are maintained in an outside system, can you remove the CastorLoginModule.

To configure an alternate (that is, other than the ICM default) authentication mode for ICM on WebSphere, follow this procedure.

To configure an alternate authentication mode for JBOSS

1 Start WebSphere.

2 Run the Administration Console, and log in as the ICM System Administrator.

3 Navigate to Security > JAAS Configuration > Application Logins view.

4 Click CE8-For-CEUserRegistry, and then click JAAS Login Modules.

At least two entries appear in the JAAS Login Modules table. One entry is for the Castor Login Module and another entry is for the Admin Login Module.

5 Click com.ibm.ws.security.common.auth.module.proxy.WSLoginModuleProxy Module Class, and then click Custom Properties of these classes.

If the value of the delegate property is `com.motiva.ce.security.spi.CastorLoginModule`, then this is the Castor Login Module.

6 Do one of the following:

- If you want both participant and administrator users authenticated by a mode other than Castor, remove the Castor Login Module and go to Step 9.
- If you want only participant users authenticated by a mode other than Castor, continue with Step 7.

7 Make sure the custom properties in the following table are present by default.

Property	Setting
delegate	com.motiva.ce.security.spi.CastorLoginModule
authenticate-administrators	true
authenticate-participants	true

8 Change the authenticate-participants property setting to false.

9 Save your changes.

10 Depending on the authentication mode you want to configure, continue to one of the following procedures:

- "Setting Up SiebelLoginModule" on page 64
- "Configuring LDAP Authentication for WebSphere" on page 64

Setting Up SiebelLoginModule

The SiebelLoginModule allows end users to be authenticated against an instance of Siebel Enterprise Server. You can set up Siebel Enterprise Server as the login module by completing the following procedure.

To set up SiebelLoginModule

1 Follow the steps of "Configuring Alternate Authentication Modes for WebSphere" on page 63.

2 Navigate to Security > JAAS Configuration > Application Logins.

3 Click CE8-For-CEUserRegistry.

4 Click JAAS Login Modules.

5 Click the first module, and then click Custom Properties.

6 Click the Delegate property.

7 Change the value of the Delegate property to com.motiva.ce.security.spi.SiebelLoginModule.

8 Add the custom properties with the values provided by Siebel server, as listed in Table 6.

Configuring LDAP Authentication for WebSphere

You can configure Siebel ICM so that it authenticates users against an LDAP server. To do this for WebSphere, after you have verified your operating system's environment settings in "Configuring Environment Settings for LDAP Authentication" on page 56, you must add a JAAS Login Module.

To add a JAAS Login Module for LDAP

1 Follow the steps of "Configuring Alternate Authentication Modes for WebSphere" on page 63.

2 To add the LDAP Login Module, perform the following steps:

a Click the JAAS Login Modules link in the bread crumbs to return to the JAAS Login Modules table.

b Make sure you have determined values for the properties as described in the following table.

Property	Description
host	Hostname of your LDAP server. Must be fully qualified domain name (<LDAP_HOSTNAME>).
port	Port number of your LDAP server (<LDAP_PORT>).
DNPrefix	uid=
DNSuffix	Check with your LDAP server administrator to determine these values (<SUFFIX>).

c Click New to add a new JAAS Login Module.

d Click the Custom Properties link, and add the custom properties listed in the following table.

Property	Setting
delegate	com.motiva.ce.security.spi.LDAPLoginModule
LdapContextFactory	com.sun.jndi.ldap.LdapCtxFactory
authentication-type	simple
host	<LDAP_HOSTNAME> For example, icmwin2k01.corp.siebel.com
port	<LDAP_PORT> For example, 15774
DNPrefix	uid=
DNSuffix	<SUFFIX> For example, ou=People,dc=corp,dc=siebel,dc=com

e Click Save.

3 Restart WebSphere.

About WebSphere Login Module Custom Properties

Table 6 describes the custom properties you can set for the WebSphere login modules.

Table 6. Custom Properties

Property	Login Module	Value	Comments
connectString	SiebelLoginModule	AsSiebelCRMProvided	Applies only to authentication with Siebel CRM applications. For the connect string syntax, see "Connect String Syntax" on page 66.
language	SiebelLoginModule	AsSiebelCRMProvided	defaultDomain is NTLM only.
multi-threaded	All	true	Applies only to authentication with Siebel CRM applications.
unauthenticatedIdentity	All	nobody	Required. Default is nobody.

Connect String Syntax

The Siebel connect string syntax is as follows:

```
siebelProtocol://gatewayServer/enterpriseServer/appObjMgr
```

Naming conventions are as follows:

```
siebelProtocol = siebel[[.transport][.[encryption][.[compression]]]]
   transport = {tcpip|http}  (default = tcpip)
   encryption = {none | rsa}  (default = none)
   compression = {none | zlib}  (default = zlib)
gatewayServer = hostname:port where server is installed. Port is 2321.
enterpriseServer = Siebel
appObjManager = XXXObjMgr_enu
```

A Upgrading from ICM 7.8 to 7.8.2

This appendix describes the process of upgrading from ICM 7.8 to 7.8.2. For assistance with implementing these procedures, please contact Siebel Professional Services. This appendix contains the following sections:

CAUTION: These procedures do *not* apply to upgrades from any release except 7.8.2, or from earlier versions of this product (Motiva) to Siebel ICM. If you want to upgrade from any release except ICM 7.8 to 7.8.2, or from any version of Motiva to any version of Siebel ICM, please contact Siebel Professional Services.

Process of Upgrading from ICM 7.8

The main areas to consider when upgrading are the database upgrade and the application upgrade. Other potential areas that may require some reconfiguration or upgrade area reporting, plan configuration, and data-level configuration with external systems.

The following high-level steps are necessary to upgrade from ICM 7.8 to 7.8.2:

1. "Upgrading the ICM Application" on page 67
2. "Upgrading the ICM Database Schemas" on page 68
3. "Rebuilding the ICM Databases" on page 71
4. (Optional) "Upgrading ICM-Third Party Applications Integration" on page 72

Upgrading the ICM Application

This section describes a step in the "Process of Upgrading from ICM 7.8" on page 67. This step is required for all upgrades.

There is no special upgrade procedure for the ICM application. You only have to install ICM version 7.8.2. For detailed instructions, see Chapter 3, "Meeting the Requirements for Installing Siebel ICM" and Chapter 4, "Siebel Incentive Compensation Management Installation."

When configuring the property files during installation of ICM version 7.8.2, use the same Host Name, Ports, and Database Information as in ICM 7.8.

It is recommended that you keep the ICM 7.8 installation and application server directories. Back up these databases before starting the installation of ICM version 7.8.2 and after the installation is complete.

CAUTION: Do *not* install ICM version 7.8.2 over ICM 7.8. Create a different <STAGING> directory for ICM version 7.8.2. Also, install a new version of the application server. Create a different <DEPLOYMENT> location so that you do not overwrite the old one.

Upgrading the ICM Database Schemas

This section describes a step in the "Process of Upgrading from ICM 7.8" on page 67. This step is required for all upgrades.

This section describes how to upgrade the transaction database and analytics database schemas. Depending on your database platform, perform one of the following procedures:

- "Upgrading the ICM Transaction Database Schema for SQL Server" on page 68
- "Upgrading the ICM Transaction Database Schema for Oracle" on page 69
- "Upgrading the ICM Transaction Database Schema for DB2" on page 69

NOTE: These instructions assume familiarity with the installation procedures for your database platform. For more information about these procedures, see the section about your database application in "Installing the Database Application" on page 20.

Upgrading the ICM Transaction Database Schema for SQL Server

To upgrade a SQL Server transaction database schema, follow this procedure.

To upgrade the ICM transaction database schema for SQL Server

1. Using isql or SQL Analyzer, log in to SQL Server as the transaction schema owner account while connected to the old transaction database that you want to upgrade.
2. Run the following script with no arguments:

   ```
   <STAGING>/db/sql/transaction/sqlserver/migrate_78GA_to_782.sql
   ```

3. Save the output of the script to the following location:

   ```
   <STAGING>/db/sql/transaction/sqlserver/migrate_78GA_to_782.log
   ```

4. Run the following script with no arguments:

   ```
   <STAGING>/db/sql/transaction/sqlserver/
   migrate_legacy_to_transforming_import_services.sql
   ```

   This upgrades the SERVICE_TYPE table, thus allowing import services that have been upgraded for 7.8.2 to process information from the database.

Upgrading the ICM Transaction Database Schema for Oracle

To upgrade the ICM transaction database schema for Oracle, follow this procedure.

To upgrade the ICM transaction database schema for Oracle

1. Using sqlplus, log in to Oracle as the transaction schema owner account of the old transaction database that you want to upgrade.
2. Run the following script:

   `<STAGING>/db/sql/transaction/oracle/migrate_78GA_to_782.sql`

   With these arguments:

   - Oracle data tablespace name
   - Oracle index tablespace name
   - Oracle DB sizing for large tables (you must specify in M, k or b)
   - Oracle DB sizing for medium tables (you must specify in M, k or b)

   For example:

   ```
   sqlplus <user>/<password>@<SID> @migrate_78GA_to_782.sql ICM_TX_DATA ICM_TX_INDEX
   2500k 1000k
   ```

3. Save the output of the script to the following location:

   `<STAGING>/db/sql/transaction/oracle/migrate_78GA_to_782.log`

4. Run the following script with no arguments:

   ```
   <STAGING>/db/sql/transaction/oracle/
   migrate_legacy_to_transforming_import_services.sql
   ```

   This upgrades the SERVICE_TYPE table, thus allowing import services that have been upgraded for 7.8.2 to process information from the database.

Upgrading the ICM Transaction Database Schema for DB2

To upgrade a DB2 transaction database schema, follow this procedure.

To upgrade the ICM transaction database schema for DB2

1. Connect to the ICM database by running the following command:

   `db2 connect to <ICM_DB>`

   For <ICM_DB>, substitute the name of your ICM database; for example, ICM314.

2 Find the LOCKTIMEOUT parameter by running the following command:

```
db2 get database configuration
```

3 Set the LOCKTIMEOUT parameter by running the following command:

```
db2 update db cfg using LOCKTIMEOUT 60
```

4 Make the changes effective by running the following command:

```
db2 disconnect current
```

5 Reconnect to the ICM database.

6 Run the following command:

```
db2 AUTOCONFIGURE using ISOLATION CS APPLY DB ONLY
```

This sets the Transaction Isolation Level, a connection-specific property that defines how records are locked, to a system default of Cursor Stability (CS). This setting is suitable for systems like ICM that use numerous short-duration locks.

7 Disconect from the database by running the following command:

```
db2 disconnect current
```

8 Restart the database manager by running the following command:

```
db2 stop dbm followed by db2 start dbm
```

9 Using a DB2 Command Window, log in to DB2 as the transaction schema owner account while connected to the ICM 7.8 transaction database that you want to upgrade.

For example:

```
db2 -t connect to <database> user <transaction schema owner> using <password>
```

10 Run the following script with no arguments, specifying @ as the command terminator:

```
<STAGING>/db/sql/transaction/db2/migrate_78GA_to_782.sql
```

For example:

```
db2 -td@ -f migrate_78GA_to_782.sql
```

11 Save the output of the script to the following file:

```
<STAGING>/db/sql/transaction/db2/migrate_78GA_to_782.log
```

12 Run the following script with no arguments:

```
<STAGING>/db/sql/transaction/db2/
migrate_legacy_to_transforming_import_services.sql
```

This upgrades the SERVICE_TYPE table, thus allowing import services that have been upgraded for 7.8.2 to process information from the database.

Rebuilding the ICM Databases

This section describes a step in the "Process of Upgrading from ICM 7.8" on page 67. This step is required for all upgrades.

This section describes how to rebuild the ICM transaction and analytics databases.

NOTE: These instructions assume familiarity with the ICM version 7.8.2 standard installation procedures. For detailed information about these procedures, see Chapter 4, "Siebel Incentive Compensation Management Installation."

To rebuild the ICM databases

1. Configure your ICM instance to use the upgraded transaction database by editing the appropriate db.properties file for your database.

   For the path and file appropriate to your database, see "Configuring ICM Database Properties" on page 37.

2. Navigate to the <STAGING>/deploy directory and run the following targets in the order listed:

   ```
   ant clean-appserver
   ant fast-deploy
   ```

3. Back up the Analytics database.

4. In the <STAGING>/deploy directory, rebuild the upgraded database by running the targets listed in the following table:

To Rebuild:	Run Targets:
Transaction schema stored procedures	`ant db-transaction-procs`
Transaction views	`ant db-transaction-views`

Analytics schema for the current period	`ant db-analytics`
	`ant db-finish`
	
	NOTE: If you want to repopulate the analytics data from previous periods, you must run the Update Analytics service later. See Step 8.

5. Start the Siebel ICM application server. See "Starting the ICM Application Server" on page 46.

6. In the <STAGING>/deploy directory, complete the upgrade by running the targets listed in the following table:

To Do This:	Run Targets:
Update the locales, dictionaries, and system data	`ant populate-all`
Create the symbol constraints	`ant db-transaction-migrate-78GA-to-782-finish`

7 Update the statistics for your database platform.

For information about how to do this, see your database application's documentation.

8 If your company used the Analytics database in ICM 7.8 and will create reports from the Analytics database in ICM version 7.8.2, run the Update Analytics service.

NOTE: When you ran the ant db-analytics command in Step 4, you erased the existing Analytics database and rebuilt the schema. Because there is no longer any data in the Analytics database, you must repopulate it by running the Update Analytics service.

For information about launching a service, see the chapter on running services in *Siebel Incentive Compensation Management Administration Guide*.

Upgrading ICM-Third Party Applications Integration

This section describes a step in the "Process of Upgrading from ICM 7.8" on page 67. This step is optional.

There may be cases (mainly involving exports from ICM) where you need to upgrade the integrations with third party systems. For example:

- If there are new entities within ICM that ICM exports to a third party system, then you must build a new data integration.
- Your company might create its own data warehouse and use a tool other than an ICM export service to extract data from ICM and send it to the data warehouse. If the ICM data model changes during an upgrade, then you must change the data export interface accordingly.

B Siebel ICM and Informatica PowerCenter

This appendix describes how to install and use Informatica PowerCenter with Siebel ICM. This appendix contains the following sections:

About Informatica PowerCenter

Informatica PowerCenter is an ETL (Extract-Transform-Load) tool that moves data from one database to another. You can install Informatica PowerCenter and set it up to work with ICM. This tool can improve the performance of the Update Analytics service, which transfers data from the ICM transaction database to the ICM Analytics database.

Informatica PowerCenter works with Siebel ICM by running a series of workflows. The workflows come with ICM and must be loaded into PowerCenter. Each workflow corresponds to a table in the Analytics database.

When you run the Update Analytics service, it calls APIs that start the workflows in Informatica PowerCenter. Each workflow loads data from the source (transactional) database, performs some transformation of the data, and then loads it into the target (Analytics) database. When the workflows have finished running, Informatica PowerCenter returns control of the process to ICM.

NOTE: The configuration to run Unicode data in this appendix is based on the Windows operating system as purchased in United States, using English as an input and display language. If you have purchased Windows in a different country, or if you use a different input and display language, some details may differ.

Server, Source and Target Code Page Relationships

A *code page* is a table that maps sequences of bits to specific characters. The code page for your system determines the characters in the server, source, and target. In this case, *source* refers to the files or database from which data is transferred, and *target* refers to the files or database to which data is transferred. If the code page is incorrect, the data may not be transferred correctly or you may lose some character data.

Operating system vendors often provide their own sets of proprietary code pages. Code pages contain the encoding to specify characters in a set of one or more languages. Within each set, different alphabets may have different code pages. Also, different languages that use the same alphabet may have different code pages.

The source's code page is a subset of the Informatica server's code page. The Informatica server's code page, in turn, is a subset of the target's code page.

The Informatica server's code page is the default code page of your operating system. If you purchase Windows in the United States and use English as an input and display language, your operating system ANSI and OEM code pages are MS Latin1 (MS1252) by default. However, if you install and use additional display or input languages, the operating system might use a different code page. For details about selecting a default code page for your operating system, and setting the input and display languages, consult the Windows documentation or contact Microsoft Technical Support.

The repository server's code page is the default code page of your operating system. The source and target code pages can be the same as your database's code pages. For more details, refer to the Informatica documentation.

Process of Setting Up Informatica PowerCenter with ICM

This section describes the tasks for installing and configuring Informatica PowerCenter, and for setting up ICM to work with Informatica PowerCenter.

Installing Informatica PowerCenter involves deploying its components: Client, Server, Repository Server, and ODBC. Depending on the requirements of your system, you may want to install these components all at once on the same machine, or individually on different machines. For detailed instructions on installing and configuring Informatica PowerCenter, see the Informatica documentation.

NOTE: It is recommended that you install Informatica PowerCenter on a machine that is accessible to the machine on which the ICM application server is deployed, and that you install JAVA API 71 on the machine that hosts the ICM application.

The process of setting up Informatica PowerCenter with ICM consists of the following tasks:

1. "Setting Up the PowerCenter Repository Server" on page 74
2. "Starting the PowerCenter Repository Server" on page 75
3. "Setting Up the Informatica PowerCenter Server" on page 75
4. "Defining the NLS_LANG System Variable (Oracle Only)" on page 78
5. "Configuring the PowerCenter Server for ICM" on page 78
6. "Defining the ICM Database Connections" on page 79
7. "Starting the Informatica Server" on page 80
8. "Deploying PowerCenter with ICM" on page 80

Setting Up the PowerCenter Repository Server

The PowerCenter repository server contains the repository, which holds mappings, transformations, and all the data. After installing the Informatica Client tools, you must install the PowerCenter repository server. You must configure the repository before you can start it. Use this procedure to install and configure the PowerCenter repository server.

In the procedures and examples that follow, the Informatica repository server directory with which you set up Siebel ICM is identified as <INFA REPOSITORY SERVER>.

To set up the PowerCenter repository server

1. Set the library path to the absolute directory path where the Repository Server is to be installed. For example, on AIX, set the library path variable LIBPATH to the repository server directory.
2. Configure the server with the pmrsconfig utility.

   You configure the connectivity, logging, and administration password information in pmrsconfig. Enter all required parameters before starting the Repository Server. This file is stored in the same directory as pmrepserver by default and writes the configuration parameters to the file pmrepserver.cfg.
3. Install and configure the database client connectivity software on the machine hosting the Repository Server.

   For more information about installing Informatica, see the installing and configuring the UNIX Repository Server section of the Informatica documentation.

Starting the PowerCenter Repository Server

Use this procedure to start the PowerCenter repository server.

To start the PowerCenter repository server

1. Log on to UNIX machine as a user who has the privilege to start the Repository Server.
2. Run the following command:

   ```
   pmrepserver [configuration_file_name]
   ```

   By default, the Repository Server uses the pmrepserver.cfg configuration file.

   A message appears indicating that the repository server has started.

   For more information about installing Informatica, see the installing and configuring the UNIX Repository Server section of the Informatica documentation.

Setting Up the Informatica PowerCenter Server

The Informatica server gets instructions from the repository and transforms the data. It can also load data to different platforms and target types. Use this procedure to install and configure the Informatica server.

To set up the Informatica server

1. Install the Informatica server.

2 Configure the Informatica server with the pmconfig utility.

   This utility writes the configuration parameters to the file pmserver.cfg. This file, by default, is stored in the directory where the PowerCenter repository server is installed.

3 Enter the license keys to run the Informatica Server with your database platforms.

4 Install and configure the native database client connectivity software on the machine hosting the Informatica Server.

   For specific database connectivity instructions, refer to your database documentation.

Importing the ICM Repository into PowerCenter

Use this procedure to import the Siebel ICM repository.

For more information about installing Informatica, see the creating repository section of the Informatica documentation.

To import the ICM repository into PowerCenter

1 Create a new database in your RDBMS where Informatica can store repository information. Give the database a descriptive name; for example, `InfaRepos`.

2 Create a new user in your RDBMS with full read/write access to the database you created in Step 1.

   Give the user a descriptive username; for example, `infauser`; and any password. Make a note of the username and password for later reference.

3 Navigate to the following directory:

   `<STAGING>/config/informaticaConfig/`

4 Locate the SiebelICMUTF8Repository.zip file.

   This file contains the ICM repository.

5 Go to the machine that hosts the <INFA REPOSITORY SERVER> and unzip the SiebelICMUTF8Repository.zip file to the following directory:

   `<INFA REPOSITORY SERVER>/backup/`

6 Start Repository Server Administration Console. From the Windows Start Menu, choose Programs > Informatica PowerCenter > Informatica PowerCenter - Client > Repository Server Administration Console.

7 Right-click Informatica Repository Servers, and select New Server Registration.

8 In the dialog box that appears after you add the repository, connect to a database. You can connect to a new database, or use an existing database that does not contain an Informatica repository.

   a Enter the repository server hostname, port number, and other necessary information.

   b Click OK.

c Right-click the repository server hostname and select Connect.

d Enter the password for the repository server administrator (for example, demo1234), and then click OK.

e Navigate to the Repository Server Administration Console and verify that the following items appear: Repositories, Activity Log, Backups, Available Packages.

9 From Repositories, right-click New Repository.

10 In the New Repository dialog box, perform the following steps:

a In the General tab, enter the repository name and select Do not create any content.

NOTE: A repository already exists under the specified database connection.

b In the Database Connection tab, select the Database Type. Use the database information created in Step 1 to complete the ConnectString, DBUser, and DBPassword fields.

NOTE: An example of the ConnectString for SQL Server DB is sqlserverHost@DBName. You can click Help to see the format of ConnectString for various databases.

c If you are using Unicode, in the Database Connection tab, in the CodePage field, select ISO 8859-1 Western European.

d When you restore a repository in UNIX, choose ISO 8859-1 Western European as your code page.

e Further, when you restore a repository in Windows, choose MS Windows Latin 1 as your code page.

CAUTION: You cannot change the code page after the repository is created.

f In the Licenses tab, enter the Product license key, and click Update.

g To verify that licenses for PowerCenter and all databases are updated, enter the Connectivity license key, and click Update.

h Click Apply, and then OK.

11 Right-click the repository (the same one used in Step 7), and go to <All Tasks> <Restore>.

12 In the Restore Repository dialog box, select the SiebelICMRepository.rep, and then click OK.

13 In the Activity log, verify that all the SQL scripts have run successfully.

14 Right-click the repository, select Start to verify that the status of the repository is shown as running.

15 If you have not done so already, configure the new repository for Informatica PowerCenter Server. For information about this task, see "Setting Up the Informatica PowerCenter Server" on page 75.

NOTE: You may need to restart the Informatica Repository server and the Informatica PowerCenter Server after the configuration.

Defining the NLS_LANG System Variable (Oracle Only)

If you use Oracle as your repository database, you must define the NLS_LANG system variable on your repository server machine. This variable indicates to Informatica the language, territory, and character set of the operating system. To define the NLS_LANG system variable, follow this procedure.

To define the NLS_LANG system variable

1. Determine the Oracle database character set.
   a. From the Repository Server machine, connect to the Repository Database using SQL*Plus.
   b. Run the following query:

      ```
      SELECT * FROM V$NLS_PARAMETERS
      ```

   c. Look for NLS_LANGUAGE, NLS_TERRITORY, and NLS_CHARACTERSET.
   d. Make a note of these values.
   e. The combination of these three is the NLS_LANG.

      The NLS_LANG format is as follows:

      ```
      <NLS_LANGUAGE>_<NLS_TERRITORY>.<NLS_CHARACTERSET>
      ```

2. Set the NLS_LANG variable for the environment of the user that starts the Repository Server as follows.
   a. In a Bourne shell, enter the following:

      ```
      $NLS_LANG=<NLS_LANGUAGE>_<NLS_TERRITORY>.<NLS_CHARACTERSET>; export NLS_LANG
      ```

   b. In a C shell, enter the following:

      ```
      $NLS_LANG <NLS_LANGUAGE>_<NLS_TERRITORY>.<NLS_CHARACTERSET>
      ```

   In these entries, use the values from the V$NLS_Parameters query.

Configuring the PowerCenter Server for ICM

Use this procedure to configure the Informatica PowerCenter server for Siebel ICM.

To configure the PowerCenter server for ICM

1. Run the pmconfig utility. Modify each entry with the appropriate information, and press Enter to go the next entry.

   If you are using Unicode, select ISO 8859-1 Western European as the Code Page.

2. From the Windows Start Menu, choose Programs > Informatica PowerCenter > Informatica PowerCenter Client > Workflow Manager.

   This opens the Workflow Manager.

3 Connect to the Repository with the Administrator password.

The default username/password is Administrator/Administrator.

4 In Workflow Manager, from the Server menu, choose Server Configuration.

a Select the required server and click Edit.

b If you are using Unicode, select the same Code Page you selected for the ICM repository.

See "Importing the ICM Repository into PowerCenter" on page 76.

c Enter the Server name and PowerCenter Server hostname.

d Click Resolve Server to resolve the IP Address of the PowerCenter Host.

e Click Advanced.

Make sure the <INFA REPOSITORY SERVER> attribute points to the directory where the PowerCenter Server is installed.

f Click OK, and then select Yes.

g In the Server Browser dialog box, click Close.

Defining the ICM Database Connections

Use this procedure to define the Siebel ICM database connections in Informatica PowerCenter. You can edit an existing connection or define a new one.

For more information about editing or adding new connections, see the section on configuring the workflow manager in the Informatica documentation.

To edit an existing ICM database connection

1 To edit the existing relational database, in Workflow Manager, do the following:

a Choose Connections > Relational Connection Browser.

b Select the database you want to edit and click Edit.

c In the popup window, enter the user name, password, connect string, code page, and server name that applies to the database type.

d Click OK.

2 If you are using Unicode, select UTF-8 encoding of Unicode as the Code Page.

To define a new ICM database connection

1 To edit the existing relational database, in Workflow Manager, do the following:

a Choose Connections > Relational Connection Browser.

b Click New.

c In the popup window, select the database type; for example, Oracle; and click OK.

d In the popup window, enter the user name, password, connect string, code page, and server name that applies to the database type.

e Click OK.

2 If you are using Unicode, select UTF-8 encoding of Unicode as the Code Page.

3 To complete the new relational database, do the following:

a In the repository in which you are defining database connections, check the folder to make sure it is not in use.

If it is in use, right click the folder and choose Disconnect from the context menu.

b From the Workflow Manager, choose Connections > Replace.

c Add two new connection pairs, and replace the default source and target connections with the ones created in Step 1.

d Click Replace, and then verify that the Output Window is updated with the log information.

Starting the Informatica Server

Use this procedure to start the Informatica Server.

To start the Informatica Server

1 Verify that the repository database, and the Repository Server managing the repository are running.

2 Log on to the UNIX machine on which the Informatica Server is running.

3 Run the following command:

pmserver [pmserver.cfg]

To verify that the Informatica Server started, check the Informatica Server event log.

NOTE: The event log is defined by the user during setup and configuration.

Deploying PowerCenter with ICM

Use this procedure to deploy Informatica PowerCenter with Siebel ICM.

To deploy PowerCenter with ICM

1 Install Informatica JavaLMAPI71 on the machine that hosts Siebel ICM application, and make sure the lib directory is in the system path.

For example, on AIX:

```
export LIBPATH=$LIBPATH:< JavaLMAPI71InstallFolder>/lib:
```

2 Navigate to the <STAGING>/deploy/informatica directory, and modify the properties in the infa.properties file, as necessary.

Some properties are described in the following table.

Property	Comments
infa.server.hostname	Name of the machine hosting the informatica server
infa.server.port	Informatica server port number
infa.repository.GID	Repository globe ID
infa.repository.name	Repository name
infa.repository.user.id	Repository username
infa.repository.user.password	Repository password
infa.repository.folder.name	Name of the folder under Repository in the Informatica Workflow Manager.
infa.api.root	The location of where JavaLMAPI71 is installed
infa.workflow.concurrent.number	Number of CPUs of the machine hosting the Informatica server

3 Stop the ICM application server.

4 Open a command prompt window, and navigate to the <STAGING>/deploy folder.

5 Run the following command:

```
ant fast-deploy
```

6 Restart the ICM application server.

7 Login to ICM, and run the Update Analytics Service.

8 Use the Workflow Monitor tool to verify the workflows.

About Siebel ICM Workflows

This section describes issues associated with ICM workflows run by the Update Analytics service that interact with Informatica PowerCenter.

Workflow and Session Logs

Workflow logs are located in the <INFA REPOSITORY SERVER>/WorkflowLogs/ directory. Session logs are located in the <INFA REPOSITORY SERVER>/SessLogs/ directory.

Viewing Workflow and Session Status

You can view the workflow and session status in real time.

To view the workflow and session status

1. Navigate to Informatica Client > Workflow Monitor.
2. Connect to the ICM Repository.
3. Connect to the Server > On Line.

   You can see the status in both the Gantt Chart and Task views.
4. To view more detail about the workflow and session, navigate to the Task view, expand the server navigation tree, and perform the following steps:
   - **Workflow detail.** Right-click the workflow, and choose Get Workflow Log.
   - **Session detail.** Right-click the session, and choose Get Session Log.

Error Messages

You may ignore the following warning messages:

- CMN_1079 WARNING
- CMN_1103 WARNING
- TE-7004 WARNING

However, all the sessions and workflows must complete successfully. Check detailed errors in each session log.

Constraints

Siebel ICM supports one Update Analytics service at one time; that is, only one Update Analytics service can be in a running status at any given time.

C Setting Up ICM Distributed Processing

This appendix describes how to set up multi-server distributed processing for each application server platform supported by ICM. It includes the following topics:

About Designing the Distributed Infrastructure

To distribute service processing, you can install Siebel ICM on multiple servers. These servers are called *processors*. Processors are grouped together by one central controller application server called the *controller*, which can also act as a processor. The controller maintains the processors' service state information, manages each service run's context information, and acts as the Trace/Log master.

This section lists design rules for setting up a distributed ICM configuration, and provides examples of distributed processing infrastructure.

Rules for All Installations

The following rules apply to JBoss and WebSphere installations:

- The controller machine can also be a processor machine, or it can be a controller machine only.
- Multiple processors can be on any machine, including the one that has the controller.
- You must assign different port numbers to the controller and to each processor that is installed on the same machine. For a controller and processors on separate machines, the port numbers can be the same.
- On a super multi-CPU machine (that is, a machine with more than two CPUs), it is recommended that you install multiple processors to get better *GC* ("Garbage Collection") performance. GC is the process by which the system recovers Java memory.
- When running a service batch that references import files on a remote drive, you must share the remote directory and configure the local machine to mount the remote drive.
- For the recommended number of RAM/CPU with respect to the number of MDB/JVM, consult your hardware platform's documentation.

Rules for WebSphere Installations

The following rules apply only to WebSphere installations:

- WebSphere's Network Deployment Manager can be on the same machine as WebSphere Application Server, or it can be on a different machine.
- Because Network Deployment Manager maintains the configuration and status of all the application servers, it is recommended that you install it on a stand-alone machine. This machine does not have to be very powerful; Windows OS with one CPU that has maximum memory of 512 MB is sufficient.

Examples of WebSphere Installations

The diagrams in this section illustrate some different types of distributed ICM installations on a WebSphere platform.

Figure 1 shows a configuration where the controller is also a processor, each processor resides on a separate machine, and Network Deployment Manager resides on a stand-alone machine.

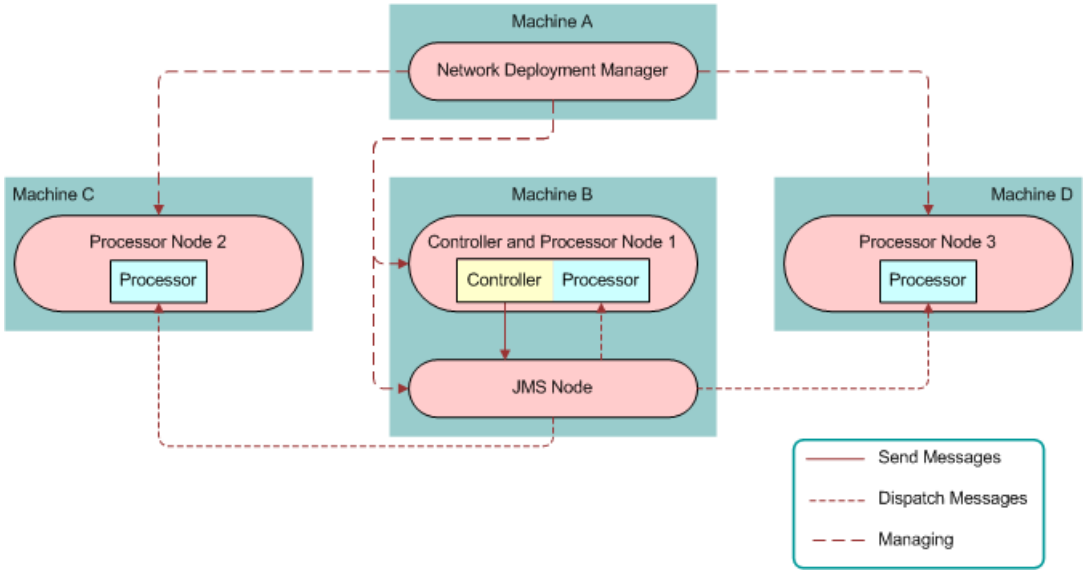

Figure 1. Distributed WebSphere Installation, Simple

Figure 2 shows a configuration where the Controller is also a processor. Multiple processors share one machine, that is the Controller resides on the same machine as a processor node and the Network Deployment Manager resides on a stand-alone machine.

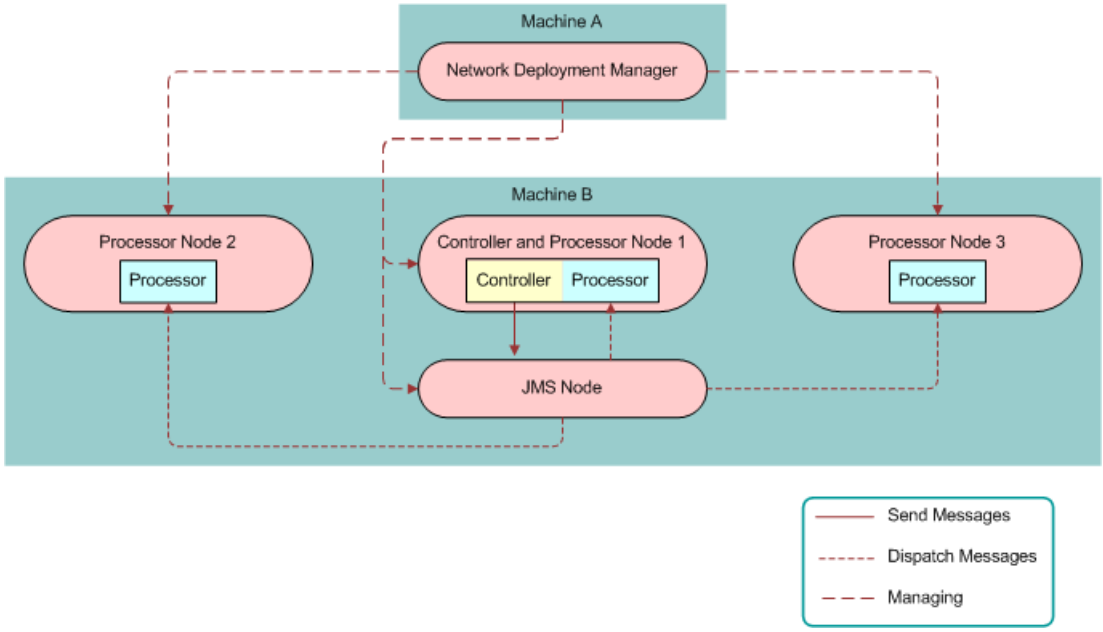

Figure 2. Distributed WebSphere Installation, Medium

Figure 3 shows a configuration where the controller is also a processor, each machine has one node, each node contains four servers, each processor resides on its own server, and the JMS provider resides on the controller node.

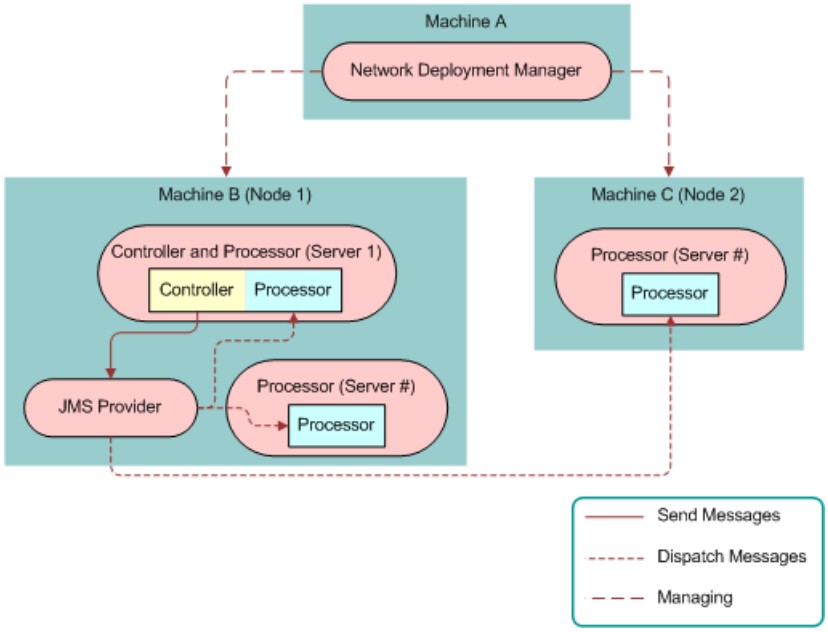

Figure 3. Distributed WebSphere Installation, Complex

Variable Directory Naming Conventions

In this appendix, variable directory names in network paths are named in the following way:

- The controller directory is labeled <CONTROLLER>.
- The WebSphere Network Deployment Manager directory is labeled <MANAGER>.
- Processor directories are labeled <PROCESSOR>.

Process of Setting Up Distributed Processing for JBoss

This section describes how to set up ICM distributed processing for a JBoss application server environment. The process of setting up ICM distributed processing on the JBoss application server platform requires the following tasks:

1. “Designing the JBoss Distributed Infrastructure” on page 87
2. “Setting Up JBoss” on page 87
3. “Creating the JBoss Staging Directories” on page 87
4. “Setting Up the JBoss Controller” on page 87

5 “Setting Up the JBoss Processors” on page 88

6 “Starting the Controller and Processor JBoss Instances” on page 88

7 “Modifying JBoss Distributed Processing” on page 88

Designing the JBoss Distributed Infrastructure

To design the distributed processing topology of Siebel ICM, follow the design rules listed in “About Designing the Distributed Infrastructure” on page 83.

Setting Up JBoss

You must install JBoss on the controller machine and on every processor machine before you can set up JBoss distributed processing. You need one JBoss installation for each processor and one JBoss installation for the controller.

To set up JBoss

1 Install jboss-3.0.3_tomcat-4.1.12 for the controller and for each processor.

2 If multiple processors reside on one machine, install multiple jboss-3.0.3_tomcat-4.1.12 instances in different locations on that machine.

Creating the JBoss Staging Directories

This procedure describes how to create the JBoss staging directories.

To create the JBoss staging directories

■ Unzip one copy of the ICM <STAGING> directory for the controller and one for each processor. If one machine has multiple processors, each processor needs an ICM <STAGING> directory.

Setting Up the JBoss Controller

This procedure describes how to create the JBoss controller.

To set up the JBoss controller

1 In the controller <STAGING> directory, change the deploy.default.properties, appserver.properties, db.properties, app.service.properties, and service.properties files to point to the properties of the controller node.

2 Run the following target:

```
ant deploy
```

3 In the controller <JBoss>/bin directory, start JBoss by running the following command:

```
run.bat
```

4 Navigate to the <STAGING>/deploy directory and run the following target:

```
ant populate-all
```

5 To confirm that the controller is set up correctly, log on to ICM and make sure that it is running.

Setting Up the JBoss Processors

This procedure describes how to create the JBoss processors.

To set up the JBoss processors

1 In each processor <STAGING>/deploy directory, change the deploy.default.properties, appserver.properties, db.properties, app.service.properties, and service.properties files to point to the properties of that processor node. These files are located in the root of <STAGING>/deploy and in its subdirectories.

2 Navigate to the <STAGING>/deploy directory and run the following target:

```
ant fast-deploy
```

3 Start JBoss by running `sh run.sh` in each processor's <JBoss>/bin directory.

Starting the Controller and Processor JBoss Instances

This procedure describes how to start the controller and processor JBoss instances.

To start the controller and processor JBoss instances

1 Start the controller's JBoss instance completely.

2 Start each processor's JBoss instance.

Starting the controller and processor JBoss instances starts the controller and processor.

NOTE: You can start all the noncontroller processors concurrently.

Modifying JBoss Distributed Processing

After the JBoss distributed processing configuration is set up, you can change it by removing or redeploying the controller or a processor.

To redeploy a JBoss controller or processor

1. Repeat the steps of "Setting Up the JBoss Controller" on page 87 and "Setting Up the JBoss Processors" on page 88.
2. To redeploy the controller, stop and restart the controller and all processors.
3. To redeploy a processor, stop and restart only that processor.

To remove a JBoss processor

- To remove a processor from the node list, stop that processor.

  If you do not stop the processor, it continues to receive messages from the controller.

Process of Setting Up Distributed Processing for WebSphere

This section describes how to set up ICM distributed processing for a WebSphere application server environment. The process of setting up ICM distributed processing on the WebSphere application server platform requires the following tasks:

1. "Designing the Distributed Infrastructure for WebSphere" on page 89
2. "Removing Previous Versions of ICM from WebSphere" on page 90
3. "Installing WebSphere Applications" on page 90
4. "Creating the WebSphere Staging Directories" on page 91
5. "Setting Up the WebSphere Controller" on page 91
6. "Setting Up the WebSphere Processors" on page 92
7. "Installing the WebSphere Network Deployment Manager" on page 92
8. "Adding the Nodes to Network Deployment Manager" on page 93
9. "Deploying the JMS Server and Node Agents" on page 95
10. "Starting the WebSphere Application Servers" on page 96
11. "Stopping the WebSphere Application Servers" on page 97
12. "Setting Up the Java Client" on page 97

NOTE: This process requires a newly installed Websphere instance on which no other applications have been previously deployed.

Designing the Distributed Infrastructure for WebSphere

To design the distributed processing topology of Siebel ICM, follow the design rules listed in "About Designing the Distributed Infrastructure" on page 83.

Removing Previous Versions of ICM from WebSphere

If you have WebSphere Network Deployment set up and a previous version of Siebel ICM is installed in the Network Deployment Manager, you must remove it before setting up the current version of ICM.

To remove previous versions of ICM from WebSphere

1. If Network Deployment Manager is stopped, start it from the <MANAGER>/bin directory.
2. Go to the Network Deployment Manager Administrative Console and remove all the Siebel applications.
3. From the <DEPLOYMENT>/bin directory, run the following command:

   `sh removeNode.sh -username <USERNAME> -password <PASSWORD>`

   For <USERNAME>, substitute your ICM administrator user name; default is `mark`. For <PASSWORD>, substitute your ICM administrator password; default is `anil`.

   This command removes the ICM nodes from the Network Deployment Manager.
4. Start each node and go to their Applications Managers to uninstall all the corresponding Siebel applications.
5. Remove all the cloned servers (if there are any).
6. Stop all the WebSphere servers.

Installing WebSphere Applications

You must install WebSphere software on all controller and processor machines.

To install WebSphere applications

1. Install one instance of WebSphere, its Cumulative Fix, its Java patch, and its MQ patch on each controller and processor machine. For information, see "Installing WebSphere" on page 25.
2. Install one instance of WebSphere Network Deployment and its Cumulative Fix on the Network Deployment Manager machine.

   It is recommended that this be a stand-alone machine.
3. Synchronize the clocks on the controller, processor, and Network Deployment Manager machines.

NOTE: There cannot be any space characters in the names of directories where the WebSphere components are installed. Also, after the installation, the WebSphere application server's bin directory must be in the env's PATH parameter.

Creating the WebSphere Staging Directories

This procedure describes how to create the WebSphere staging directories.

To create the WebSphere staging directories

1. Unzip one copy of the ICM <STAGING> directory for Network Deployment Manager into the <MANAGER> directory on the Network Deployment Manager machine.
2. Unzip one copy of the ICM <STAGING> directory for the controller into the <CONTROLLER> directory on the Controller machine.
3. Unzip one copy of the ICM <STAGING> directory for each processor into each <PROCESSOR> directory on the processor machines.

Setting Up the WebSphere Controller

This procedure describes how to create the WebSphere controller.

To set up the WebSphere controller

1. In the controller <STAGING>/deploy directory, change the deploy.default.properties, appserver.properties, db.properties, app.service.properties, and service.properties files to point to the properties of the controller node. These files are located in the root of <STAGING>/deploy and in its subdirectories.
2. Run the following target:

   ```
   ant deploy
   ```

3. Start the controller's WebSphere application.
4. Navigate to the <STAGING>/deploy directory and run the following target:

   ```
   ant populate-all
   ```

5. (Optional) From the <STAGING>/deploy directory, run the following target:

   ```
   ant populate-module -Dmodule=<MODULE>
   ```

   For information about the syntax of this target, see "populate-module -Dmodule=<MODULE>" on page 123.

6. To confirm that the controller is set up correctly, log on to ICM and browse through the ICM application UI to see whether the controller is functioning.

   Displaying the ICM UI confirms that the JNDI is binding correctly, the JDBC configuration is functioning correctly, and the Web application is deployed successfully. If issues exist for any of these, the ICM login page does not appear.

Setting Up the WebSphere Processors

This procedure describes how to create the WebSphere processor. You can set up a processor on the same node as the controller or on a different node.

To set up a WebSphere processor on a controller node

1. If the controller's WebSphere instance is not running, start it.
2. Start the WebSphere Single Server Administrative Console.
3. In the WebSphere Single Server Administrative Console, clone another server (for example, server2) from the default application server template by performing the following steps:
   a. Navigate to Servers > Application Servers.
   b. Click New.
   c. Follow the steps to create a new server from the default application server template, and accept all the default settings.

   For more information, see the WebSphere documentation.
4. In each of the processor <STAGING>/deploy directories, change the deploy.default.properties, appserver.properties, db.properties, app.service.properties, and service.properties files to point to the properties of the processor server.
5. Navigate to the <STAGING>/deploy directory and run the following target:

   ```
   ant fast-deploy
   ```

To set up a WebSphere processor on a noncontroller node

1. In each of the processor <STAGING>/deploy directories, change the deploy.default.properties, appserver.properties, db.properties, app.service.properties, and service.properties files to point to the properties of the processor server.
2. In each appserver.properties file, change the deploy.processor.app.name property to a name that is unique in your network deployment environment.
3. Navigate to the <STAGING>/deploy directory and run the following target:

   ```
   ant fast-deploy
   ```

Installing the WebSphere Network Deployment Manager

The Network Deployment Manager is a WebSphere tool that manages the nodes of a WebSphere network.

To install the WebSphere Network Deployment Manager

1. In the Deployment Manager machine's ICM <STAGING>/deploy directory, change the deploy.default.properties, appserver.properties, and appserver.manager.properties files to point to the properties of the Network Deployment Manager node.

   NOTE: In the appserver.properties file, change only the deploy.appserver.root and deploy.appserver.SAS.home parameters, to point to Network Deployment Manager installation directory.

2. Navigate to the <STAGING>/deploy directory and run the following target:

   ```
   ant deploy-network-manager
   ```

3. Run the following command:

   ```
   startManager.sh
   ```

   This starts the Deployment Manager in the <DEPLOYMENT_MANAGER_ROOT>/bin directory.

4. Browse through the Network Deployment Administrative Console to make sure the ICM version of Deployment Manager is set up correctly.

   Because global security is enabled, the login window appears.

5. Log in as the ICM administrator.

   The default administrator user name is `mark`, and the default administrator password is `anil`.

   NOTE: You can change this default ICM user name and password through WebSphere. For information, see Changing the ICM Username and Password for WebSphere on page 51.

Adding the Nodes to Network Deployment Manager

After installing the WebSphere Network Deployment Manager, you add the WebSphere nodes so the Network Deployment Manager knows they exist.

To add the nodes to Network Deployment Manager

1. Start the Deployment Manager in the <DEPLOYMENT_MANAGER_ROOT>/bin, if it is not already started, by running the following target:

   ```
   startManager.sh
   ```

2. Find out the <SOAP_CONNECTOR_ADDRESS>. In the Administrative Console, navigate to System Administration > Deployment Manager > End Points > <SOAP_CONNECTOR_ADDRESS>.

3. Navigate to each processor's <APPLICATION SERVER>/bin directory, and run the following command in each one of them:

```
sh addNode.sh <MANAGER_HOST> <SOAP_CONNECTOR_ADDRESS> -includeapps -noagent -
username <USERNAME> -password <PASSWORD>
```

For <USERNAME>, substitute your ICM administrator user name; default is mark. For <PASSWORD>, substitute your ICM administrator password; default is anil.

4 In the Administrative Console, find out the environment's virtual hosts by performing the following steps:

   a Navigate to Environment > Virtual Hosts > admin_host > Host Aliases.

   b Navigate to Environment > Virtual Hosts > default_host > Host Aliases.

5 Confirm that the controller server's and the Deployment Manager Console's HTTP transport port numbers are the ones defined in the environment's virtual hosts.

   If they are not the same, change them by performing the following steps:

   a Navigate to Servers > Application Servers > <SERVER> > Web Container > HTTP Transport.

   b Change the controller server's HTTP transport port numbers to make them consistent with the environment's virtual hosts.

   c Navigate to System Administration > Deployment Manager > HTTP Transport.

   d Change the Deployment Manager Console HTTP transport port number to make it consistent with the environment's virtual hosts.

6 Confirm that the controller and processor servers' JNDI port numbers are the ones you defined in the appserver.properties.

   When you add a node, WebSphere can potentially change port numbers. If the port numbers are not the ones you defined, perform the following steps:

   a Navigate to Servers > Application Servers > <SERVER> > End Points > BOOTSTRAP_ADDRESS.

   b Change the JNDI port numbers to the ones defined in appserver.properties file.

7 Check the JMS nodes, Network Manager node and all the node agents' JNDI port numbers to make sure they are *not* the ones you defined in all the processors within this machine.

   If a conflict exists, change the JNDI port number to a unique number by completing one of the following steps:

   - For JMS nodes, navigate to JMS servers > each jmsserver > End Points > BOOTSTRAP_ADDRESS.
   - For the Network Manager node, navigate to System Administration > Deployment Manager > End Points > BOOTSTRAP_ADDRESS.
   - For Node agents, navigate to System Administration > node agents > each nodeagent > End Points > BOOTSTRAP_ADDRESS.

Deploying the JMS Server and Node Agents

The Java Messaging Service (JMS) Server is a stand-alone server that sends messages asynchronously. This allows asynchronous processing, which makes ICM distributed processing possible.

To deploy the JMS server and node agents

1. Navigate to the <STAGING>/deploy/websphere5.1/ directory.
2. Open the appserver.manager.properties file in a text editor, locate the properties that point to the controller's base server, and change them as described in the following table.

Property	Comments
deploy.baseserver.node.name	Change each property to point the base server to the controller server.
deploy.baseserver.server.name	
deploy.baseserver.root	
deploy.baseserver.cell.name	Change this property to point to the cell name to which all the nodes are being added. In this case, that is the cell name of the Deployment Manager.

3. Navigate to the <STAGING>/deploy directory and run the following target:

   ```
   ant deploy-controller-node
   ```

4. For each WebSphere node where ICM processors reside, open the appserver.manager.properties file in a text editor and change the properties as described in the following table.

Property	Comments
deploy.baseserver.node.name	Change each property to point the base server to one processor server.
deploy.baseserver.server.name	
deploy.baseserver.root	

   NOTE: The deploy.baseserver.cell.name property takes the same setting for the controller and all the processors.

5. Navigate to the <STAGING>/deploy directory and run the following targets:

   ```
   ant deploy-processor-node
   ```

   ```
   ant deploy-processor-jms
   ```

6. Repeat Step 4 and Step 5 for each additional WebSphere node where ICM processor/processors reside.
7. Navigate to each node's bin directory, and run the following command:

```
syncNode.sh <MANAGER_HOST> <SOAP_CONNECTOR_ADDRESS> -username <USERNAME> -password
<PASSWORD>
```

For <USERNAME>, substitute your ICM administrator user name; default is mark. For <PASSWORD>, substitute your ICM administrator password; default is anil.

8 If the processor is on a different node from the controller, change the JMS Queue Factory on each processor node to point to the controller JMS server if they are not pointing there already, using the following steps:

   a Navigate to Resources > Websphere JMS Provider.

   b Choose a node, and click Apply.

   c Navigate to Websphere Queue Connection Factories > Siebel Queue Connection Factory.

   d Change the Node, Component-managed Authentication Alias, and Container-managed Authentication Alias to point to the controller node's values.

9 If the processor is on a different node from the controller, change the JMS Topic Factory on each processor node to point to the controller JMS server if they are not pointing there already, using the following steps:

   a Navigate to Resources > Websphere JMS Provider.

   b Choose a node, and click Apply.

   c Navigate to Websphere Topic Connection Factories > Topic Connection Factory.

   d Change the Node, Component-managed Authentication Alias, and Container-managed Authentication Alias to point to the controller node's values.

NOTE: In the WebSphere distribution setup, do not run any ant target that makes changes conflicting with the changes from Network Deployment Manager.

Starting the WebSphere Application Servers

This procedure describes how to start the WebSphere application servers.

To start the WebSphere application servers

1 Make sure the Network Deployment Manager is up and running. If it is not, navigate to the Network Deployment Manager bin directory, and run the following command:

```
startManager.sh
```

2 Go to each node's bin directory, and run the following command:

```
startNode.sh
```

3 Open the Network Deployment Manager Administrative Console, and navigate to System Administrator > Node Agents.

4 In the Node Agents manager, make sure all the node agents have the status Started.

5 Navigate to servers > JMS server, and check the JMS server status.

6 If the JMS server is not started, start it from the Network Deployment Manager Administrative Console.

7 In the Network Deployment Manager Administrative Console, navigate to servers > Application Servers, and start the controller server.

8 After controller server is fully started, in the Deployment Manager Administrative Console, navigate to servers > Application Servers, and start all the processor servers.

Stopping the WebSphere Application Servers

You can stop the controller and all the processors at the same time.

To stop the WebSphere application servers

1 Navigate to servers > Application Servers, and stop the processor servers.

2 Navigate to servers > JMS server, and stop the JMS servers.

3 Navigate to System Administrator > Node Agents, and stop the node agents.

4 Navigate to the Network Deployment Manager bin directory, and run the following command:

```
stopManager -username <USERNAME> -password <PASSWORD>
```

For <USERNAME>, substitute your ICM administrator user name; default is mark. For <PASSWORD>, substitute your ICM administrator password; default is anil.

This command stops the Network Deployment Manager.

Setting Up the Java Client

This procedure allows the controller to handle pure Java client requests such as Service Launcher, Populators, State Manager Snapshot, and so on.

To set up the Java client

■ Navigate to the <STAGING>/deploy directory and run the following target:

```
ant deploy-controller-client-in-multi-server-mode
```

D Siebel ICM Report Utility

This appendix describes the Siebel ICM Report Utility installation and configuration. It includes the following topics:

About the ICM Report Utility

The Siebel ICM Report Utility allows users to create, test, and publish Jasper Reports using JasperReports. *JasperReports* is an open-source Java reporting tool that is designed to create page-oriented, ready-to-print documents. JasperReports can deliver dynamic content to the screen; to a printer; or to PDF, HTML, XLS, CVSV, and XML files. It is written entirely in Java and can be used in a variety of Java-enabled applications, such as J2EE or Web applications.

Reports can run against either the transaction database or the analytics database. The Report Utility can operate as a stand-alone utility, or it can be integrated with a Siebel ICM installation. Siebel ICM can run reports without installing the Report Utility. However, to create, test, and publish reports, you must install the Report Utility.

The Siebel ICM Report Utility provides the following functions:

- Compiling Jasper Report .xml files into Jasper Report .jasper files
- Filling Jasper Report .jasper files with data and creating .jrprint files
- Rendering the .jrprint files in either .pdf or .html format
- Displaying the rendered .pdf or .html files
- Publishing (copying and renaming) .jasper files from the working directory to the deployment directory for use in a dashboard

NOTE: Throughout this appendix, the Siebel ICM install directory is referred to throughout this guide as <STAGING>. The application server installation directory is referred to as <DEPLOYMENT>.

For more information on how to build Jasper Reports, consult the *JasperReports Ultimate Guide* at the following URL:

`http://jasperreports.sourceforge.net/more.docs.html`

Process of Installing the ICM Report Utility

The procedures in this section explain how to install the ICM Report Utility. Follow the procedures that are appropriate for your system, in the order listed.

1 "Installing Java Developers Kit" on page 100

This task is required for stand-alone installations that use the ICM Report Utility.

2 "Installing Apache Ant" on page 100

This task is required for stand-alone installations that use the ICM Report Utility.

3 "Installing JasperReports" on page 101

This task is required for all installations that use the ICM Report Utility.

4 "Installing Siebel ICM" on page 101

This task is required for integrated installations that use the ICM Report Utility.

5 "Installing ICM Report Utility" on page 101

This task is required for all installations that use the ICM Report Utility.

6 "Installing a Database Driver for ICM Report Utility" on page 102

This task is required for installations with a DB2 or Oracle database that use the ICM Report Utility.

Installing Java Developers Kit

This procedure applies to stand-alone installations that use the ICM Report Utility.

Siebel ICM requires the Java Developers Kit (JDK) to operate. Use the following procedure to install the Java Developers Kit.

This task is a step in "Process of Installing the ICM Report Utility" on page 100.

To install the Java Developers Kit

- Follow the steps of "Installing Java Developers Kit for JBoss" on page 28.

Installing Apache Ant

This procedure applies to all installations that use the ICM Report Utility.

Apache Ant allows you to run Ant target commands, which are used throughout the ICM Report Utility installation process.

This task is a step in "Process of Installing the ICM Report Utility" on page 100.

To install Apache Ant

- Follow the steps of "Installing Apache Ant" on page 31.

Installing JasperReports

This procedure applies to stand-alone and integrated installations that use the ICM Report Utility.

Use this procedure to install the JasperReports utility.

NOTE: The JasperReports utility is only available in deprecated mode for existing customers. It is not supportable for new customers, who must use Actuate.

This task is a step in "Process of Installing the ICM Report Utility" on page 100.

To install JasperReports

1. Download jasperreports-0.5.2-project.zip from the following location:

   `http://prdownloads.sourceforge.net/jasperreports/jasperreports-0.5.2-project.zip?download`

2. Install the product in the default directory or in another directory of your choice.

   NOTE: This installation directory is referred to throughout this guide as <JASPERREPORTS>.

Installing Siebel ICM

This procedure applies to integrated installations that use the ICM Report Utility.

This task is a step in "Process of Installing the ICM Report Utility" on page 100.

To install Siebel ICM

- Follow the installation and configuration instructions in Chapter 4, "Siebel Incentive Compensation Management Installation."

Installing ICM Report Utility

This procedure applies to all installations that use the ICM Report Utility.

The Siebel ICM Report Utility is included with Siebel ICM.

This task is a step in "Process of Installing the ICM Report Utility" on page 100.

To install the ICM Report Utility

1. Locate ReportUtility-7.7.zip in your Siebel ICM distribution.

2 Install the product in the default directory or in another directory of your choice.

NOTE: This installation directory is referred to throughout this guide as <REPORT UTILITY>.

Installing a Database Driver for ICM Report Utility

This procedure applies to installations with a DB2 or Oracle database that use the ICM Report Utility.

The Siebel ICM Report Utility ships with a driver for SQL Server. However, for DB2 or Oracle, the driver must be downloaded and installed separately.

This task is a step in "Process of Installing the ICM Report Utility" on page 100.

To install a database driver for DB2 or Oracle

1 Locate the driver library for your database installation.

2 Perform either of the following steps, as appropriate to your database type:

- For DB2, copy the driver library JAR or ZIP file to <REPORT UTILITY>/lib/db/db2.
- For Oracle, copy the driver library JAR or ZIP file to <REPORT UTILITY>/lib/db/oracle.

Process of Configuring the ICM Report Utility

The procedures in this section explain how to configure the ICM Report Utility, and how to publish and test reports. Follow the procedures that are appropriate for your system, in the order listed.

1 "Configuring ICM to Use the Report Utility" on page 103

This task is required for integrated installations that use the ICM Report Utility.

2 "Configuring the Report Utility" on page 103

This task is required for all installations that use the ICM Report Utility.

3 "Configuring Database Settings for ICM Report Utility" on page 105

- "Configuring Stand-Alone Database Settings" on page 106
- "Configuring Integrated Database Settings" on page 106

This task is required for all installations that use the ICM Report Utility. Follow the procedure appropriate to your installation's integration status.

4 "Customizing Reports for an ICM Installation" on page 106

This task is required for integrated installations that use the ICM Report Utility.

5 "Configuring Security for ICM Report Utility" on page 107

This task is required for integrated installations that use the ICM Report Utility.

6 “Publishing Reports with ICM Report Utility” on page 108

This task is required to make Jasper Reports available to administrators.

7 “Testing a Report” on page 109

This task is required to verify that Jasper Reports run correctly.

Configuring ICM to Use the Report Utility

This procedure applies to integrated installations that use the ICM Report Utility.

This task is a step in “Process of Configuring the ICM Report Utility” on page 102.

To configure Siebel ICM to use the Report Utility

1 Navigate to <STAGING>/deploy.

2 Open the deploy.default.properties file.

3 Edit the deploy.report.utility.root property to point to your <REPORT UTILITY> directory.

Configuring the Report Utility

This procedure applies to all installations that use the ICM Report Utility.

Follow this procedure to configure the ICM Report Utility to run on your system.

This task is a step in “Process of Configuring the ICM Report Utility” on page 102.

1 Navigate to <REPORT UTILITY>/deploy.

2 Open the report.properties file and edit the properties as shown in Table 7.

Table 7. Siebel ICM Report Utility Properties

Property	Required for	Description
report.jasper.reports.root	Publishing and Testing	Location of the JasperReports Project Home. Typical location: /home/jasperreports.
report.db	Publishing and Testing	Type of database to use. Options include: sqlserver, oracle, db2.
deploy.analyticsDB.host	Publishing and Testing	Machine name of the computer hosting the analytics DB. Typical setting: localhost.
deploy.transactionDB.host	Publishing and Testing	Machine name of the computer hosting the transaction DB. Typical setting: localhost.
report.java.home	Testing Only	Location of the Java developers kit. Typical setting: /usr/java.

Table 7. Siebel ICM Report Utility Properties

Property	Required for	Description
report.browser	Testing Only	Location of the browser executable used for viewing PDF or HTML report output. Typical setting: `c:\Program Files\Internet Explorer\iexplore.exe`
report.temp.dir	Publishing and Testing	Folder to which the source .xml files are copied (report.temp.dir/src) and then compiled to .jasper files (report.temp.dir/compiled). Typical location: temp/deploy.
report.src.dir	Publishing and Testing	Location of the Jasper Report definition (.xml) files. Typical location: <STAGING>/report/JasperReports.
report.publish.dest.dir	Publishing and Testing	Run-time deployment directory of the application server to which compiled reports are copied. Typical location: <DEPLOYMENT>/report.
report.publish.master.report	Publishing Only	Identifies a top-level report, which may contain subreports. Maps that report to an operating unit. For information about variables for this property, see "Report Utility Properties File Variables" on page 105.
report.name	Testing Only	Name of the report that you want to test. This value should match both the .xml file name and the <jasperReport> tag's name attribute in the .xml file. For example: report.name=ParticipantPayoutStatement
report.paramN	Testing Only	Specifies parameter values to pass into a report. When testing a report, you must pass parameter values into the report. Specify the parameters to be passed by replacing N with a sequential number that uniquely identifies the parameter. For information about parameters for this property, see "Report Utility Properties File Variables" on page 105.

Report Utility Properties File Variables

This section lists variables and their values for certain properties in the report.properties file.

report.publish.master.report

Specify one report.publish.master.report variable for each OU/Report combination. Although you may have multiple subreport files, you only have to define the report.publish.master.report variable for each top-level report made available to the users. The value of the variable is:

```
<OU_Code>$<REPORT_NAME>
```

For example, to make the ParticipantPayoutStatement report available to the TIPMSOU and OU_B1 operating units, add the following two lines:

```
report.publish.master.report =OU_B1$ParticipantPayoutStatement
report.publish.master.report =TIPMSOU$ParticipantPayoutStatement
```

NOTE: OU Codes must not contain the dollar sign ($) character.

report.paramN

When a report is in production, the application server passes the parameters that follow to every report. Therefore, it is strongly recommended that you also pass these parameters to your report during testing:

- **sys_participant_code.** Code of the participant. For a participant report, this is the participant code for the user who is currently logged on. For a Manager report or an Executive report, this is usually entered by the manager as a prompt value.
- **sys_PeriodNumber.** Value of the absolute working period. This value is derived from the user's current working period.
- **sys_EnterpriseUnitCode.** Code for the user's Enterprise Unit.
- **sys_OperatingUnitCode.** Code for the user's active Operating Unit.
- **sys_participant_user_uuid.** user.UUID of the user running the report. For a participant report, this is the user.UUID for the user who runs the report. For a Manager or Executive report, this is the user.UUID of the manager or executive who runs the report.

A valid report.paramN value is a name=value pair. For example:

```
report.param1=sys_participant_code=02646
report.param2=sys_PeriodNumber=16
report.param4=sys_EnterpriseUnitCode=0016
report.param5=sys_OperatingUnitCode=0001
report.param6=sys_participant_user_uuid=02646
```

Configuring Database Settings for ICM Report Utility

You use different procedures to configure database settings for the Report Utility if it is running as a stand-alone utility and if it is integrated with Siebel ICM.

This task is a step in "Process of Configuring the ICM Report Utility" on page 102.

Configuring Stand-Alone Database Settings

Use this procedure to configure database settings for a stand-alone ICM Report Utility Installation.

To configure stand-alone database settings

1. Navigate to <REPORT UTILITY>/deploy/<DATABASE>/.
2. Launch a text editor, and open the db.properties file.
3. Read the comments for each property, and set the property accordingly.

Configuring Integrated Database Settings

Use this procedure to configure database settings for a ICM Report Utility Installation that is integrated with Siebel ICM.

To configure integrated Report Utility database settings

1. Navigate to <STAGING>/deploy.
2. Run the following command:

   ```
   ant report-apply-db-properties
   ```

   This copies the database properties from the Siebel ICM instance to the Report Utility.

Customizing Reports for an ICM Installation

This procedure applies to integrated installations that use the ICM Report Utility.

Some preconfigured reports shipped with Siebel ICM require customization to work with the custom reading-type fields defined for each operating unit.

To customize a report for an ICM Installation

1. Navigate to the report.src.dir directory that you set in the ICM Report Utility's report.properties file.

   See "Configuring the Report Utility" on page 103.

2. Make a copy of the Jasper .xml file for each operating unit that needs access to that report.
3. With an XML or text editor, change the operating unit-specific reading types to match your installation.

For example, to customize the Payout Statement, open the <STAGING>/report/JasperReports/PayoutStatment.xml file and replace these values with the equivalent values for your OU installation, as shown in the following table.

Report Parameter	Custom Field	Comments
PROMPT_GOAL_RDNG1	Loan Amount	Used in goal profiles.
PROMPT_CREDIT_RDNG1	Loan Amount	Used in credit profiles.
PROMPT_MEASURE_CODE	Loan Volume	Measure code used for performance evaluation.

Configuring Security for ICM Report Utility

This procedure applies to integrated installations that use the ICM Report Utility.

You can configure Siebel ICM Report Security to map security roles to report groups and reports to report groups.

To configure report security for ICM Report Utility

1. Navigate to <STAGING>/report/security and open report_groups.txt.

   The report_groups.txt file defines groups of reports for display in the Report menu.

2. Define the report groups available to each Operating Unit.

   Each line in report_groups.txt defines a group of reports for a single Operating Unit. The file is a comma-delimited list with the following column headers:

   ```
   <OUCode>.<Group Name>, ReportCode1, ReportCode2, ReportCodeN
   ```

   For example, the entries that follow define one report group for two OUs.

   ```
   TIPMSOU.Payment, ParticipantPayoutStatement
   OU_B1.Payment, ParticipantPayoutStatement
   ```

3. Open report_acl.txt.

   The report_acl.txt file maps the report groups defined in report_groups.txt to operating unit-specific security roles.

4. Define the report groups available to each role.

   Each line in this file maps an Operating Unit Role to a Report Group.

   NOTE: OU codes and Role names must match the exact values stored in the transaction database. In addition, the Group codes must match the exact values defined in report_groups.txt.

   The file is a comma-delimited list with the following column headers:

```
<OUCode>.<Siebel ICM Security Role Code>, GroupCode1, GroupCode2, GroupCodeN
```

For example, the entries that follow map the roles for the TIPMSOU and OU_B1 Operating Units to the Payment report group.

```
TIPMSOU.Participant, Payment
TIPMSOU.Manager, Payment
TIPMSOU.ADMIN ROLE: TIPMSOU, Payment
TIPMSOU.ManagerDashboard, Payment
OU_B1.Participant, Payment
OU_B1.Approver1, Payment
OU_B1.Approver2, Payment
OU_B1.AccountMgr, Payment
OU_B1.SalesRep, Payment
OU_B1.ExecutiveDashboard, Payment
```

5 Navigate to <STAGING>/deploy, and run the following command:

```
ant deploy-report-security
```

This command copies the security files from the <STAGING> directory to the report production environment.

Publishing Reports with ICM Report Utility

This procedure applies to all installations that use the ICM Report Utility.

To make a report available for use, you must publish it.

This task is a step in "Process of Configuring the ICM Report Utility" on page 102.

To publish reports with ICM Report Utility

1 Make sure you completed the publication properties in the report.properties file. See Table 7 on page 103 for entries that have the Publishing value in the Required column.

2 Add the JDK\bin directory (for example, /usr/java/bin) to the system path.

CAUTION: If you do not put this entry in the system path, the report compilation process fails.

3 Navigate to <REPORT UTILITY>/deploy and run the following target:

```
ant report-compile-all
```

4 If the system displays any error messages, open the appropriate .xml file, correct the error, and recompile by rerunning the following target:

```
ant report-compile-all
```

5 Publish the reports by running the following target:

```
ant report-publish-all
```

This target copies the *.jasper files in the <REPORT_SOURCE> directory to the <REPORT_PUBLISH_DESTINATION> directory. Then, it renames each master report file in this directory according to the report.publish.master.report settings you specified in the report.properties file, using the naming convention that follows.

```
<OUCode>$<REPORT_NAME>.xml
```

6 Restart the application server and navigate to the dashboards to view the reports.

Testing a Report

This procedure applies to all installations that use the ICM Report Utility.

After you configure the ICM Report Utility and publish the reports, you can test them in the environment to which you published.

This task is a step in "Process of Configuring the ICM Report Utility" on page 102.

1 Make sure you completed the testing properties in the report.properties file.

See Table 7 on page 103 for entries that have the Testing value in the Required column.

2 Fill in all the report.properties required for testing.

See Table 7 on page 103.

3 Navigate to <REPORT UTILITY>/deploy, and enter the commands listed in the following table.

Ant Command	Comments
`ant report-compile-all`	First time only. Required so that subreports get compiled. Compiles all the reports in the <REPORT_SOURCE> directory.
`ant report-compile`	Compiles the report specified by the report.name property in the report.properties file.
`ant report-run-html`	For the report specified by the report.name property in the report.properties file, runs the report SQL, fills the report with data, and writes a .jrprint file. Displays the resulting HTML file in a Web browser.

TIP: You can run multiple ant commands at the same time by specifying them in a single command; for example, you can specify:

```
ant report-compile report-run-html
```

E Configuring Actuate iServer for Siebel ICM

This appendix describes how to install and configure Actuate iServer and Siebel ICM for reporting. This chapter contains the following sections:

About Actuate iServer

The Actuate iServer is a report server that handles report generation, scheduling, and security. The Actuate Management Console installs with the Actuate iServer. You can set up Actuate and ICM to work together to generate ICM reports.

For more information about Actuate applications, see the Actuate documentation.

Process of Setting Up Actuate iServer and ICM for Reporting

This section describes the tasks for installing and configuring Actuate iServer, and for setting up Siebel ICM to work with an Actuate iServer.

The process of setting up an Actuate iServer to work with Siebel ICM includes the following tasks.

Installing a Database Client on the Actuate iServer Machine

Install a client instance of your database application on the computer where you install the Actuate iServer. You must do this before installing the Actuate iServer. For information on installing a database client, see the installation section of your database application's documentation.

Installing the Actuate iServer

There is no specific machine or network location where you have to install the Actuate iServer to work with Siebel ICM. There is also no interdependence between Actuate and the other third-party applications used by Siebel ICM. The Actuate iServer can be installed before, after, or during the process of installing other third-party applications. Furthermore, you can install the Actuate iServer according to your own preference and your company's network architecture guidelines. For specific instructions on how to install an Actuate iServer, see the installation section of the Actuate iServer documentation.

It is recommended that you install the Actuate iServer before installing Siebel ICM.

However, there are possible exceptions:

- Your company may want to get Siebel ICM up and working without reporting capabilities before installing the Actuate iServer. If you do this, then after installing the Actuate iServer, you must rerun the deploy-config target to deploy the reporting.properties file. This file, and the steps for running deploy-config, are described in "Configuring ICM for Actuate Reporting" on page 114. For additional information about the deploy-config target, see "Targets for Installation" on page 121.
- Your company may want to use a pre-existing Actuate iServer (in other words, an Actuate iServer already being used by other applications prior to ICM installation).

Adding a Partition on the Actuate iServer

Siebel ICM requires a dedicated partition on the Actuate iServer. This section provides a general description of the steps for setting up the partition. For specific instructions on how to set up a partition on an Actuate iServer, see the Actuate iServer documentation.

To add a partition on the Actuate iServer

1. Log in to the Actuate iServer as the System Administrator.
2. Navigate to System Partitions > Add Partition.

3 Complete the fields as shown in the following table.

Field	Comments
Partition name	Name of the partition that contains the reports volume for ICM. You can define the partition name according to your preferences and your company's network architecture guidelines.
Partition Path	Network directory path to he partition that contains the reports volume for ICM. You can define the partition path according to your preferences and your company's network architecture guidelines.

4 Click OK.

Adding an ICM Volume on the Actuate iServer

Siebel ICM requires its own volume on the Actuate iServer. This section provides a general description of the steps for setting up the volume. For specific instructions on how to set up a volume on an Actuate iServer, see the Actuate iServer documentation.

To create a new volume for ICM on the Actuate iServer

1 Log in to the Actuate iServer as the System Administrator.

2 Navigate to System Volumes > New Volume, and click the General tab.

3 Complete the fields as needed.

Some fields are described in the following table.

Field	Comments
Volume name	Name of the volume that contain the ICM reports. You can define the Volume name according to your preferences and your company's naming conventions. This value is included later in the reporting.properties file.
Description	Description of the volume that contains the ICM reports.
Primary partition	Name of the partition that contains this volume.
Transaction log path	Network directory path to the transaction log file.

4 Click Apply.

5 Click the Server Assignments tab.

6 From the Available servers section, select the server on which you want to put the volume, and click OK.

7 Click the Partitions tab.

8 To start the partition, perform the following steps:

   a From the Available Partitions section, select the partition you want to use for the volume.

   b Click the arrow to move your selection to the Selected Partitions section.

   c Select the partition in the Selected Partitions section, and then click Start.

   d Click OK

9 Select the volume and choose Take Online from the context menu.

   The status of the volume changes to ONLINE.

Creating the ICM System User on the Actuate iServer

After creating the ICM volume on the Actuate iServer, you must create an ICM System user for that volume. Your ICM application uses the ICM System user to perform a variety of reporting functions, including loading and running report executables.

To create the ICM System user

1 Log out of System Administration, and log in to the new volume as administrator for that volume.

2 Navigate to Users > Create User.

3 Create a user to act as the ICM System user.

   The recommended value is `SiebelICM`.

   You can leave all other user fields blank.

4 (Optional) Set a password for this user.

   The recommended value is `siebel2004`.

   NOTE: After the Actuate Security Extension is activated, the password is not necessary.

Configuring ICM for Actuate Reporting

To configure Siebel ICM for reporting, you must modify the properties in the reporting.properties file. The properties in this file specify the network location of the Actuate iServer, as well as other information necessary for Siebel ICM to locate and use the Actuate iServer.

To configure ICM for Actuate reporting

1 Navigate to the `<STAGING>/deploy/actuate` directory.

2 Open the reporting.properties file in a text editor, and modify the properties as described in the following table.

Property	Comments
reporting.actuate.host	Name of the machine on which the Actuate iServer is running.
reporting.actuate.port	Actuate iServer connection port.
reporting.actuate.root	Root directory used by ICM on the Actuate iServer. The recommended value is SiebelICM. The reporting.actuate.root property defines the folder name on the Actuate server, which is used to store the ICM-related report files. This folder can be created manually if desired. However if it is not already present, it is created automatically by ICM when a report executable is uploaded.
reporting.actuate.volume	Actuate iServer volume used by ICM. See "Adding an ICM Volume on the Actuate iServer" on page 113.
reporting.system.user	Name of the Siebel ICM System user on the Actuate iServer. See "Creating the ICM System User on the Actuate iServer" on page 114.
reporting.system.password	Password used to authenticate the Siebel ICM System user on the Actuate iServer. See "Creating the ICM System User on the Actuate iServer" on page 114.
reporting.actuate.allow.admin.login	If set to true, allows the volume administrator to log in to the Actuate Management Console after the ICM Actuate security extension has been installed. If set to true, you must set the reporting.actuate.admin.user and reporting.actuate.admin.password properties.
reporting.actuate.admin.user	User name of the volume administrator for the volume specified in reporting.actuate.volume. Required only if reporting.actuate.allow.admin.login is set to true.
reporting.actuate.admin.password	Password of the volume administrator for the volume specified in reporting.actuate.volume. Required only if reporting.actuate.allow.admin.login is set to true.

3 After the file is configured, restart the application server as described in the steps that follow.

NOTE: If you have configured the reporting.properties file before installing Siebel ICM, then you do not need to restart the application server.

a Stop the ICM application server.

b From the<STAGING>/deploy directory, run the following target:

```
ant deploy-config
```

For more information about the `ant deploy-config` target, see "Target Commands Reference" on page 121.

c Restart the ICM application server.

Installing the ICM and Actuate Security Extension for UNIX

The Siebel ICM and Actuate Security Extension is used to perform external authentication of Actuate users through Siebel ICM. The following steps describe how to install and configure the ICM Actuate Security Extension on an AIX or Solaris Actuate iServer. In the following steps, <ACTUATE_ROOT> refers to the folder where the Actuate iServer is installed.

To install the Actuate Security Extension for AIX

1 Navigate to the ActuateSecurityExtension folder in the ICM distribution, and then locate the ActuateSecurityExtension-7.8.2.1.zip file.

2 Unzip the ActuateSecurityExtension-7.8.2.1.zip file to a temporary location.

3 Navigate to the \aix subdirectory of the unzipped location, and locate the rsse.so and ICMActuateBridge.class files.

4 Copy the rsse.so and ICMActuateBridge.class files to the following directory on the machine on which the Actuate iServer is installed:

```
<ACTUATE_ROOT>/AcServer/RSSE/SiebelICM
```

5 In a text editor, open the .profile for the user account used to start the Actuate server, and add the following environment variables:

```
ICM_BRIDGE_CLASSPATH=/<ACTUATE_ROOT>/AcServer/RSSE/SiebelICM
ICM_SERVER_URL=http://<HOSTNAME>:<PORT_NUMBER>
export ICM_BRIDGE_CLASSPATH
export ICM_SERVER_URL
```

For <HOSTNAME>, substitute the host name of the machine running the ICM application server. For <PORT_NUMBER>, substitute the port number on which ICM is running; for example, 8080.

6 In the .profile, add the JVM location to the LIBPATH environment variable by adding the following lines:

```
LIBPATH=$LIBPATH:/<ACTUATE_ROOT>/AcServer/jdk141/jre/bin:/<ACTUATE_ROOT>/AcServer/
jdk141/jre/bin/classic

export LIBPATH  (if not already present)
```

7 In the .profile, verify that the JAVA_HOME variable is set. If it is not set, add the following line:

```
JAVA_HOME=<ACTUATE_ROOT>/AcServer/jdk141
```

8 In the .profile, verify that <JAVA_HOME>/bin is part of the PATH variable. If it is not, add <JAVA_HOME>/bin to the PATH variable.

To install the Actuate Security Extension for Solaris

1 Navigate to the ActuateSecurityExtension folder in the ICM distribution, and then locate the ActuateSecurityExtension-7.8.2.1.zip file.

2 Unzip the ActuateSecurityExtension-7.8.2.1.zip file to a temporary location.

3 Navigate to the /solaris subdirectory of the unzipped location, and locate the librsse.so file.

4 Copy the librsse.so file to the following directory on the machine where the Actuate iServer is installed:

<ACTUATE_ROOT>/AcServer/RSSE/SiebelICM

5 In a text editor, open the .profile for the user account used to start the Actuate server, and add the following environment variables:

```
ICM_SERVER_IP_ADDRESS=<ADDRESS>
ICM_SERVER_PORT=<PORT_NUMBER>
export ICM_SERVER_NAME
export ICM_SERVER_PORT
```

For <ADDRESS>, substitute the IP address of the ICM application server host. For <PORT_NUMBER>, substitute the port number on which ICM is running; for example, 8080.

Configuring Open Security on the Actuate iServer

This section describes how to configure open security on the Actuate iServer. Open security is a security feature which prevents users from logging into the encyclopedia volume unless the ICM application server is started. When the application server is up and running, if reporting.actuate.allow.admin.login is set to true, then the volume administrator can log into the volume.

This task is a step in "Installing the ICM and Actuate Security Extension for UNIX" on page 116.

To configure Actuate iServer Open Security

1 Log in to System Administration on the Actuate iServer.

2 Navigate through System Volumes to the ICM volume you want to configure.

3 Click the Open Security tab.

4 Select the Enable Open Security check box.

5 Set the RSSE library name to the full path and file name of the rsse library file (rsse.so for AIX or librsse.so for Solaris) that you copied in "Installing the ICM and Actuate Security Extension for UNIX" on page 116.

6 Set RSSE multithread safe to true.

7 Click Apply.

8 Restart the Actuate iServer.

Configuring the Database Connections for ICM Reports

To configure the database connections for Actuate iServer with Siebel ICM reports, you perform the following tasks:

1 "Creating the Connection Configuration File" on page 118

2 "Testing the Connection Configuration File in Actuate" on page 119

3 "Configuring the Actuate iServer to Use the Connection Configuration File" on page 119

Creating the Connection Configuration File

A *connection configuration file* is an XML file that defines one or more database connections. A connection configuration file can point to any database platform supported by Siebel ICM. Using a connection configuration file allows reports to be moved between ICM instances without changing the report design. The Actuate Connection Configuration file defines the database connections used by Siebel ICM reports with the Actuate iServer.

A sample connection configuration file, ICMConnectionConfiguration.xml, is included with the Siebel ICM distribution. This sample connection configuration file includes examples of how to define connections to the SQL Server, Oracle, and DB2 databases.

For each platform, a ConnectOptions element is used to define a database connection. In this configuration file, the transaction and Analytics databases each require their own ConnectOptions element. The Siebel ICM ConnectOptions element defines the connection for the transaction database. The SiebelICM_DW ConnectOptions element defines the connection for the analytics database. To reference your specific database, you modify the ServerName, UserName, and Password parameters.

To create the connection configuration file

1 On your Siebel ICM distribution media, locate the ActuateSecurityExtension-7.8.2.1.zip file.

2 Extract the ICMConnectionConfiguration.xml file to a location of your choice.

3 Open the ICMConnectionConfiguration.xml file in a text editor.

4 Depending on your database platform, complete one of the following instructions:

- **SQL Server.** Modify the ServerName, UserName and Password parameters to reference your databases. Do not change the DllPath property. Use the same UserName and Password as those used by Siebel ICM to connect to the databases.
- **DB2.** Modify the DataSource, UserName, and Password parameters to reference your databases. Do not change the DllPath property.

  NOTE: You must install the DB2 client on all machines that use this Connection Configuration file, including the machine that runs the Actuate iServer.

- **Oracle.** Modify the HostString, UserName, and Password parameters to reference your databases.

  NOTE: You must install the Oracle client on all machines that use this Connection Configuration file, including the machine that runs the Actuate iServer.

5 Save your changes and exit from the ICMConnectionConfiguration.xml file.

Testing the Connection Configuration File in Actuate

After you have modified the Connection Configuration file, verify that your Actuate iServer can use it.

To test the connection configuration file

1 Launch Actuate e.Report Designer Pro.

2 On the application-level menu, choose View > Options, and select the General tab.

3 Set the Configuration File property to your ICMConnectionConfiguration.xml file.

4 Test the connection configuration file by running any reports that use the defined connections, including the Siebel ICM-delivered reports.

For detailed instructions and additional information, see the Actuate e.Report Designer Pro documentation.

Configuring the Actuate iServer to Use the Connection Configuration File

To configure your Actuate iServer to use your connection configuration file, you must first determine whether the Actuate iServer is already using a connection configuration file.

To configure the Actuate iServer to use the connection configuration file

1 Log in to the Actuate Management Console as the system administrator.

2 Click the Servers tab.

3 From the Servers drop-down list, select the ICM Actuate iServer.

4 Click the Advanced tab.

5 Check the value of the Configuration File for Connections property to determine whether your Actuate iServer is using a connection configuration file.

If your Actuate iServer is already using a connection configuration file, that file is specified as the value of the Configuration File for Connections property.

If your Actuate iServer is not using a connection configuration file, then the Configuration File for Connections property has a null value.

6 Perform either of the following steps:

- If your Actuate iServer is already using a configuration file, add the ConnectOptions defined in your connection configuration file to the specified file.

  NOTE: A Connection Configuration file can contain any number of uniquely named ConnectOptions elements.

- If your Actuate iServer is not using a configuration file, update the Configuration File for Connections property to point to your connection configuration file.

7 Restart the Actuate iServer to implement your changes.

For detailed instructions and additional information, see the section on using the Actuate Management Console in the Actuate documentation.

ICM Reports Postinstallation Tasks

After you install the Actuate iServer, and set up the Actuate iServer and ICM to work together, you must load report executables into the Actuate iServer and ICM. After that, you can use ICM's reporting functionality to configure, organize, and display reports. For information about performing these tasks, see the chapter on reports in *Siebel Incentive Compensation Management Administration Guide*.

F Target Commands Reference

This appendix lists and describes Ant target commands. The syntax of these commands is:

```
ant <target>
```

The targets are divided into several classifications, as described in the following topics:

Targets for Installation

The target commands for installation are described in this section.

deploy

Deploys Siebel ICM to the application server and re-creates the database.

deploy-db

Before entering this target, you must set the database properties. See "Configuring ICM Database Properties" on page 37.

Does the following for Transaction items:

- Creates new Transaction database, drops old database if it exists.
- Creates Transaction database schema.
- Creates Transaction OS Workflow tables.
- Creates Transaction views.
- Creates Transaction seed data.
- Creates Transaction DB stored procedures.

db-transaction-procs

Before entering this target, you must set the database properties. See "Configuring ICM Database Properties" on page 37.

Use this command for upgrades. Rebuilds the transaction DB stored procedures.

deploy-appserver

Copies Web assets, libraries, and mapping files.

deploy-config

Copies the /config files from the <STAGING> directory to the application server (<DEPLOYMENT>) directory.

deploy-logging

Copies log settings from the <STAGING> directory to the application server (<DEPLOYMENT>) directory. Also copies debug or release library files, depending on the deploy.logging.mode setting. The debug setting copies libraries that generate line numbers when printing errors to the log, which is helpful in debugging. The release setting copies libraries that do not print line numbers and, consequently, run faster.

fast-deploy

Deploys Siebel ICM to the application server, but does not re-create the database.

clean-appserver

Removes Siebel ICM from the application server. On JBoss, completely removes the JBoss application server directory. On WebSphere, removes most, but not all, Siebel ICM files. Run this command before reinstalling Siebel ICM on the same instance of WebSphere.

unzip-appserver

Unzips a new application server (JBoss only). Requires setting the following <STAGING>/deploy/JBoss3.0 properties:

```
deploy.appserver.zip.src
deploy.appserver.zip.dest
```

Targets for Data Population

The target commands for data population are described in this section.

populate-all

Populates the system with locales and dictionaries.

populate-module -Dmodule=<MODULE>

Fills an ICM instance with the plan data defined in a populator file. This syntax works for two modules, populate-blink and populate-daytona. <MODULE> represents a populator.<MODULE>.properties file in the <STAGING>/etc/populate/ directory.

For example, to populate the UniversalFinance dataset, use the following command:

ant populate-module -Dmodule=UniversalFinance

The command in this example populates an ICM instance with the data defined in the following file:

<STAGING>/etc/populate/populator.UniversalFinance.properties

populate-blink

Shortcut for creating the sample Blink plan. It uses the populate-module target to create the Blink plan.

The Blink plan is demo data. The username is blinkou1 and the password is demo1234.

populate-daytona

Shortcut for creating the sample Daytona plan. The Daytona plan is used by most of the unit tests.

populate-dictionaries

Registers new symbol dictionary values from a data file into the database. Such registration is often necessary after an upgrade, to identify new symbols.

Targets for Distributed Services

The target commands for distributed services are described in this section.

Before entering a target in this section, you must set the distributed service properties.

deploy-network-manager

Configures the server as the Network Deployment Manager.

deploy-controller-node

Configures the application server as a controller.

deploy-processor-node

Configures the application server as a processor of service items.

deploy-processor-jms

Configures the application server on the processor machine as the WebSphere JMS provider.

Targets for Service Manager

The target commands for the Service Manager services are described in this section.

config-service-log

Configures the priority level for the Service Logger.

state-manager-snapshot

Interacts with the State Manager (part of the service execution framework). Also displays or clears State Manager's in-memory state. The dump option, when used with this target, writes details of services processing to the Console.

Targets for Siebel ICM Report Utility

The target commands for the Siebel ICM Report Utility are described in this section. These commands deal with creating and deploying Jasper Reports to ICM.

The Report Utility must be installed and configured according to the instructions in Appendix D, "Siebel ICM Report Utility." The targets described in this section are shortcuts to the commands available when you use the Report Utility directly.

report-apply-db-properties

Configures the report utility to use the same db.properties as this Siebel ICM installation.

report-clean

Deletes all files generated by the Report Utility in the <REPORT_PUBLISH_DESTINATION> directory.

report-compile-all

Compiles all the .xml report files in the directory specified by the report.compile.src.dir property in the report.properties file.

report-compile

Compiles the report specified in the report.properties file.

report-fill

Fills (runs queries) for the specified report and the parameters specified in the report.properties file, and produces a .jrprint file.

report-pdf

Reads the .jrprint file for the report specified in the report.properties file and exports it to PDF. You *must* run report-compile and report-fill before running this task.

report-html
Reads the .jrprint file for the report specified in the report.properties file and exports it to HTML. You *must* run report-compile and report-fill before running this task.

report-run-pdf
Compiles, fills, and exports the report specified in the report.properties file. Launches a browser to view the resulting PDF.

report-run-html
Compiles, fills, and exports the report specified in the report.properties file. Launches a browser to view the resulting HTML.

report-publish-all
Requires a successful report-compile-all.

Publishes all the compiled reports (report.src.dir/*.jasper files) to the directory specified in the report.publish.dest.dir property in the report.properties file.

Targets for Application Server Control

The target commands for controlling the application server are described in this section.

Before entering a target in this section, you must set the application server properties. See "Configuring ICM Application Server Properties" on page 39.

start-appserver
Starts the application server.

stop-appserver
Stops the application server.

stop-appserver-no-fail
Stops the application server without reporting failures.

Targets for Backup

The target commands for backup are described in this section. These targets create backups that you can send to Siebel Technical Support for analysis if the system has issues.

backup-deploy-logs

Creates a backup of the log files generated by the deployment of Siebel ICM, which should be sent to Siebel Technical Support when filing an SR about installation issues. Creates a ZIP file in the following directory:

```
<STAGING>/backup/deploy-logs
```

backup-logs

Creates a backup of the log files generated by the appserver and Siebel ICM, which should be sent to Siebel Technical Support when filing an SR. Creates a ZIP file in the following directory:

```
<STAGING>/backup/logs
```

backup-properties

Creates a backup of the properties files for Siebel ICM, which should be sent to Siebel Technical Support when filing an SR. Creates a ZIP file in the following directory:

```
<STAGING>/backup/properties
```

backup-reports

Creates a backup of the Jasper report files for Siebel ICM, which should be sent to Siebel Technical Support when filing an SR about Jasper Reports. Creates a ZIP file in the following directory:

```
<STAGING>/backup/reports
```

backup-all

Executes all the backup commands to create several ZIP files, which should be used prior to upgrades, or for severe field issues. Creates ZIP files as defined for each backup target.

backup-config

Creates a Zip file backup of the configuration properties in the following directory:

```
<STAGING>/backup/config
```

backup-all

Runs all backup commands and creates the Zip files in the following directory:

```
<STAGING>/backup
```

Targets for Other Actions

The target commands for actions other than the ones covered in the previous sections are described in this section.

use-precompiled-jsps

Swaps the web.xml file to use the one with all the precompiled jsp servlet mappings. Runs only if the build was assembled with the following command:

```
assembly.include.precompiled.jsps=true
```

Using precompiled JSP files improves UI performance by reducing the initial load time for JSP files. Edits made to JSP files are not displayed.

CAUTION: Making edits to JSP files is *not* recommended.

use-regular-jsps

Swaps the web.xml file to use the one with none of the precompiled jsp servlet mappings.

Using regular JSP compilation means that JSP files are compiled on demand. This results in a slower initial page response time. Edits made to JSP files are displayed.

CAUTION: Making edits to JSP files is *not* recommended.

hash-text

Creates an MD5 hash of the specified text. Use the following command line option:

```
-Dtext=sometext
```

encrypt-text

Encrypts the specified text for use as an encrypted username or password. Use the following command line option:

```
-Dtext=sometext
```

script-migration

Before entering this target, you must set the database properties. See “Configuring ICM Database Properties” on page 37.

Used only during upgrades from releases prior to Siebel ICM 7.5.3. Converts existing DB scripts to use SYS_****** instead of MOTIVA_****** variables.

Index

D

J

L

M

N

O

P

R

S

T

U

W

X