Siebel

Developing and Deploying Siebel CRM

July 2022

Siebel
Developing and Deploying Siebel CRM

July 2022

Part Number: E99704-02

Authors:

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

If this is software or related documentation that is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, the following notice is applicable:

U.S. GOVERNMENT END USERS: Oracle programs (including any operating system, integrated software, any programs embedded, installed or activated on delivered hardware, and modifications of such programs) and Oracle computer documentation or other Oracle data delivered to or accessed by U.S. Government end users are "commercial computer software" or “commercial computer software documentation” pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, the use, reproduction, duplication, release, display, disclosure, modification, preparation of derivative works, and/or adaptation of i) Oracle programs (including any operating system, integrated software, any programs embedded, installed or activated on delivered hardware, and modifications of such programs), ii) Oracle computer documentation and/or iii) other Oracle data, is subject to the rights and limitations specified in the license contained in the applicable contract. The terms governing the U.S. Government’s use of Oracle cloud services are defined by the applicable contract for such services. No other rights are granted to the U.S. Government.

This software or hardware is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications that may create a risk of personal injury. If you use this software or hardware in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy, and other measures to ensure its safe use. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software or hardware in dangerous applications.

This software or hardware and documentation may provide access to or information about content, products, and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services unless otherwise set forth in an applicable agreement between you and Oracle. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services, except as set forth in an applicable agreement between you and Oracle.

The business names used in this documentation are fictitious, and are not intended to identify any real companies currently or previously in existence.

Contents

Preface

This preface introduces information sources that can help you use the application and this guide.

Using Oracle Applications

To find guides for Oracle Applications, go to the Oracle Help Center at *http://docs.oracle.com/*.

Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the *Oracle Accessibility Program website*.

Contacting Oracle

Access to Oracle Support

Oracle customers that have purchased support have access to electronic support through My Oracle Support. For information, visit *My Oracle Support* or visit *Accessible Oracle Support* if you are hearing impaired.

Comments and Suggestions

Please give us feedback about Oracle Applications Help and guides! You can send an email to: *oracle_fusion_applications_help_ww_grp@oracle.com*.

1 What's New in This Release

What's New in This Release

This chapter tracks the changes in the documentation. It includes the following topics:

- *What's New in Developing and Deploying Siebel CRM Guide, Siebel CRM 22.7 Update*
- *What's New in Developing and Deploying Siebel CRM Guide, Siebel Innovation Pack 2017*

What's New in Developing and Deploying Siebel CRM Guide, Siebel CRM 22.7 Update

The following table lists the changes in this revision of the documentation to support this release of the software.

Topic	Description
Building a Network Image *About the Siebel Design Repository*	Modified topics. The bullets about *Siebel Tools* have changed (since Workflow Process and Task UI are now Workspace-enabled).

What's New in Developing and Deploying Siebel CRM Guide, Siebel Innovation Pack 2017

The following table lists the changes in this revision of the documentation to support this release of the software.

Topic	Description
About This Guide	New chapter. The Overview chapter provides a high-level overview of the process of developing and deploying a Siebel CRM enterprise.
Installing and Configuring Siebel CRM	New chapter. This chapter provides details on how to install and configure a Siebel CRM enterprise.
Basic Organization Setup	New chapter. This chapter provides an overview of basic organization setup for Siebel CRM.
Customization in Siebel CRM	New chapter. This chapter provides information about customizing in Siebel CRM.
Adding Additional Languages	New chapter. This chapter provides overview information about how Siebel CRM handles working with multiple languages in the same Siebel CRM instance and adding additional languages to Siebel CRM.

2 About This Guide

About This Guide

This guide, Developing and Deploying Siebel CRM, provides a roadmap for installing, configuring, and deploying a Siebel CRM enterprise. As a roadmap, it places the various tasks needed to deploy Siebel CRM into the context of the overall deployment process.

This guide supplements other documentation on the Siebel CRM Bookshelf and makes many references to those guides where additional detail is provided.

Related Guides

- *Configuring Siebel Business Applications*
- *Siebel Installation Guide for Microsoft Windows*
- *Siebel Installation Guide for UNIX*
- *Configuring Siebel Open UI*
- *Siebel Fundamentals Guide*
- *Siebel Deployment Planning Guide*
- *Using Siebel Tools*
- *Siebel Application Deployment Manager Guide*
- *Siebel Applications Administration Guide*
- *Siebel Database Upgrade Guide*
- *Siebel Database Upgrade Guide for DB2 for z/OS Deploying Siebel Open UI*
- *Siebel Developer's Reference*
- *Siebel Global Deployment Guide*
- *Siebel Object Interfaces Reference*
- *Siebel Object Types Reference*
- *Testing Siebel Business Applications*

3 Installing and Configuring Siebel CRM

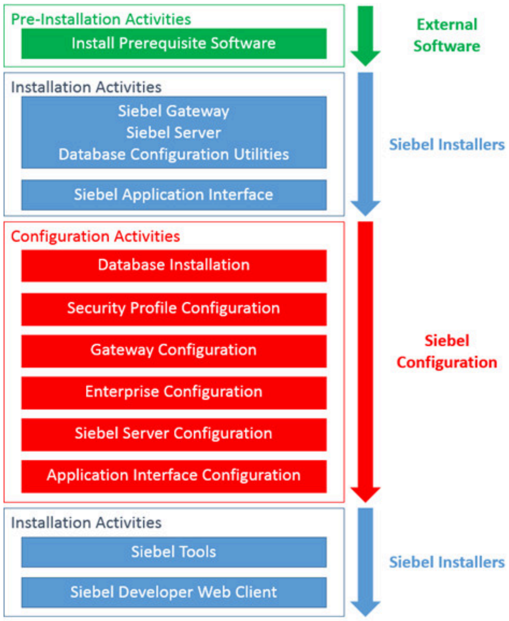
Installing and Configuring Siebel CRM

This chapter provides guidance on installing and configuring Siebel CRM. The information in this chapter is is intended as a high-level overview only. More detailed information on the information provided in this chapter can be found in *Siebel Installation Guide* , available on Siebel Bookshelf. This chapter includes the following topics:

- *Installing and Configuring Siebel CRM Roadmap*
- *Installing Prerequisite Software*
- *Downloading Siebel CRM*
- *Building a Network Image*
- *Secure Sockets Layer Requirements*
- *Installing Software for the Siebel CRM Application*
- *Creating the Siebel Database*
- *Configuring the Siebel Gateway*
- *Configuring the Siebel Enterprise*
- *Configuring Siebel Servers*
- *Configuring the Siebel Application Interface*
- *Adding the License Keys*
- *Setting up the Siebel Tools Client*
- *Setting up the Developer Web Client*

Installing and Configuring Siebel CRM Roadmap

The following image illustrates the steps required to install Siebel CRM.



As shown in this image, the steps to install and configure Siebel CRM are divided into three areas as follows:

- **Pre-installation Activities.** The tasks required to prepare your environment for Siebel CRM installation. For more information see, the following topics:

  Installing Prerequisite Software

  Downloading Siebel CRM

  Building a Network Image

  Secure Sockets Layer Requirements

- **Installation Activities.** The tasks required to install Siebel CRM on your server(s). For more information, see the following topics:

  Installing Software for the Siebel CRM Application

  Setting up the Siebel Tools Client

  Setting up the Developer Web Client

  Use Siebel installers to install the following: Siebel Tools, Siebel Developer Web Client.

- **Configuration Activities.** The tasks that prepare the various components of the software to work together to deliver the Siebel User Interface to users, interface with other systems, and so on. For more information, see the following topics:

  Creating the Siebel Database

  Configuring the Siebel Gateway

  Configuring the Siebel Enterprise

  Configuring Siebel Servers

  Configuring the Siebel Application Interface

  Adding the License Keys

Prerequisite Software

Siebel CRM installs most of the software that it requires to run. However, depending on your operating system and database, you may need to install the following software before you install Siebel CRM. The minimum software requirements for installing Siebel CRM are as follows:

- **Operating System.** A number of different operating systems support Siebel CRM, including various releases of Microsoft Windows and UNIX. Before beginning installation, confirm that your operating system is certified on the Certifications tab on My Oracle Support.
- **Enterprise Database.** A number of different enterprise databases support Siebel CRM, including various releases of Oracle, Microsoft SQL Server, and IBM DB2. Before beginning installation, confirm that your enterprise database is certified on the Certifications tab on My Oracle Support. You must install the enterprise database software before you install Siebel CRM.

  > **Note:** It is strongly recommended that you install the database platform on a separate server to that of your Siebel CRM installation.

- **Database Connectivity.** The server on which you install Siebel CRM installation requires database connectivity software to interact with the enterprise database. For example, to interact with an Oracle database, you must install the Oracle 32-bit database client. To interact a Microsoft SQL Server instance, you must install the SQL Server Native Client on each server where you install Siebel CRM. Before installing, confirm that your database connectivity software is certified on the Certifications tab on My Oracle Support.
- **Java Development Kit.** You must install the Java Development Kit (JDK) on at least one server to generate certificate requests and to install the resulting signed certificate on each server hosting any of the Siebel Enterprise Applications. JDK is freely available from the Oracle Technology Network. Before installing, confirm for specific supported versions on the Certifications tab on My Oracle Support. The 64-bit version of the JDK is required for Windows servers.

  > **Note:** Other tools are available for generating certificate requests. Your organization may choose to use one of these alternate tools, which would eliminate the need for the JDK to be installed.

- **Java Runtime Environment.** When you download Siebel CRM, the installation files are stored in a set of Java Archives (JAR) files. In order to create an image of the software during installation, you require a Java Runtime Environment (JRE). The JRE is freely available from http://www.java.com. Before installing, confirm for specific supported versions on the Certifications tab on My Oracle Support. Note that either the 32-bit or 64-bit version of the JRE is acceptable for Siebel CRM installation.

> **Note:** The JRE is only required on the machine where you generate the Siebel installation image.

Downloading Siebel CRM

You can download Siebel CRM from the following locations:

- **Oracle Software Delivery Cloud (OSDC).** The most recent Innovation Pack (IP) for Siebel CRM is available on the Oracle Software Delivery Cloud (OSDC). This is the foundation installation for Siebel CRM.
- **My Oracle Support (MOS).**The most recent Innovation Pack and subsequent Monthly Updates are available on My Oracle Support (MOS). Only customers with a current support agreement have access to MOS.

> **Note:** It is always recommended that customers use the most current Monthly Update.

To find a release on MOS, do the following:

1. Click the Patches & Updates tab.
2. In the Patch Search pane, click the Product or Family (Advanced) tab.
3. In the Product field, enter Siebel CRM.
4. In the Release is field, select two values as follows:
    - Siebel Innovation Pack 17.0
    - Siebel Innovation Pack YY.0 (where YY is the current two digit year for example 18)
5. (Optional) In the Platform is field, select the required platform.

    If you do not select a platform, you will still be able to select the download for your preferred platform on the next screen, but choosing it now will filter the search in advance.
6. Click Search button.

    The results pane displays the base Innovation Pack and available Monthly Updates.
    - The base release Innovation Pack will be labeled 17.0.0.0 23073
    - Monthly Updates will be shown in the format: YY.MM.0.0 PATCHSETYY_MM, such as 18.5.0.0 PATCHSET18_5.
        - YY represents the two digit year in which the Update was released.
        - MM represents the month (one or two digits) in which the Update was released.
7. Download the base release, for example, 17.0.0.0 for your required Innovation Pack.

    When prompted to use Download Manager, it is recommended to click Yes.
8. Download the latest Monthly Update.

    Monthly Updates are always cumulative. If there is more than one Update available, then you only need to download the latest. You must save the Update to a separate folder from the folder in which you saved the Innovation Pack files.

    > **Note:** It is normal for the base Innovation Pack release to have more files than the Monthly Update release files because the Update installation contains only incremental changes from the base release.

Building a Network Image

This topic describes creating a network image.

> **Note:** The syntax here is shown for Windows servers; there are very minor differences if you create a network image on a UNIX-based system.

After you download the installation files, you need to build a network image for both the base version and the latest Update. You then execute the network image on each machine where you want to install Siebel CRM. However, the files that you download cannot be installed directly, and you are required to extract these files before installation. This is done using the Siebel Network Image Creator (SNIC). You must perform the following steps for the base release file and again for the patch set file. If you are planning to install the application on a UNIX system, you can create the network images on a Windows system, or vice versa. Creating a network installation image is a separate process from performing the installation.

1. In Windows Explorer, navigate to the base release files that you downloaded from Oracle Software Delivery Cloud (OSDC) or My Oracle Support (MOS). For example: `c:\SLC11XHQ\Downloads\17`.
2. Unzip the base release files into the same folder.

   The following diagram illustrates the correct file structure after you unzip the base release files.
3. Open a command or shell window and navigate to the path where you unzipped the files.
4. Ensure that the JAVA_HOME variable is set to the location of your Java Runtime Environment (JRE), for example: `set JAVA_HOME=C:\Program Files (x86)\Java\jre1.8.0-151`.
5. Launch the snic.bat (Windows) or snic.sh (UNIX) script to start building the network image.
6. When prompted to generate the network image, you can accept the defaults, however, consider the following important points:
   - **Platform.** AIX, HPUX, Linux, Solaris, and Windows are available. If you know you will be installing Siebel CRM components on both Windows and Solaris boxes, for example, you can select both at this time.
   - **Products.** There are five choices available. The minimum requirement is:
     - **Siebel Enterprise Server.** Required for all Siebel CRM environments.
     - **Siebel Tools.** This is required for functionality that is unavailable in Siebel Web Tools.
     - **Siebel Web Client.** Required for script debugging, workflow simulation, and other development tasks.
   - **Languages.** Siebel offers 22 standard languages. At a minimum, it is strongly recommended that you select English (ENU), even if it is not your intended primary language. If there is any possibility that you require more languages in Siebel CRM, it is recommended that you create images for them now and install all potential languages during your initial installation. It is more convenient to do this now, rather than adding languages later.

   On completion, the SNIC prepares the installer image and saves the installation files in the location you specified.
7. Repeat this procedure for the Monthly Update files that you downloaded from MOS.

Secure Sockets Layer Requirements: Overview

Several Siebel Application components are hosted within application containers that need to trust each other and also need to be trusted by client browsers or other external servers that integrate with Siebel CRM. This requirement is accommodated by creating a Secure Sockets Layer (SSL) certificate.

There are three steps required to create a SSL certificate:

- Generate a certificate request on the server.
- Send the certificate to a certificate authority (CA) to approve the request and issue the certificate.
- Import the certificate into a keystore used by the Siebel CRM components.

> **Note:** For more information on SSL requirements, see *Siebel Security Guide* .

About Keystores

Secure Sockets Layer (SSL) certificates and Trusted Root Certification Authorities are placed in keystores on each server where Siebel CRM components are installed.

Siebel CRM uses the following keystores:

- **Keystore.** This contains the private key and your SSL certificate, in addition to the certificates for the certificate authority (CA) which issues your SSL certificate. The certificate in this keystore is used for securing:
  - Inter-server communication.
  - Communication between your browser and other applications for inbound requests to Siebel CRM.
- **Truststore.** This contains at least a root certificate for the CA which issues your SSL certificate and it may contain additional Trusted Root Certificates. The certificate in this keystore is used for the following purposes:
  - Inter-server communication between Siebel CRM components. This is accommodated by the certificate from the CA which issues your SSL certificate.
  - Outbound integration. When Siebel CRM makes outbound calls to other services, it uses the root certificates in this file to verify the identity of other services. For example, consider a call originating in Siebel CRM to a Google API. Google's SSL certificate is issued by an intermediate authority (Google Internet Authority G2) whose certificate was issued by the Trusted Root Certification Authority, GeoTrust. In order to trust the Google certificate, Siebel CRM must first trust GeoTrust, and therefore requires that the GeoTrust root CA certificate be imported into the Truststore.

Generating a Certificate Request

Regardless of the certificate authority (CA) type that you plan to use, you need to generate a certificate request. A simple way to do this is with the keytool utility included in the Java Development Kit (JDK) as shown in the following example.

The following procedure assumes that the JDK has been installed and that the bin folder is in the current path.

> **Note:** There are other tools available to generate certificate requests; the following procedure is one option available to you.

1. Open a command prompt window.
2. Execute the following commands:

    a. **x:**

       Change to the drive where you plan to install Siebel, for example *C:* or *D:*. (Windows only).

    b. **cd \**

       Makes sure that you are at the root of the folder. For Linux/Unix, select the appropriate base path where you will eventually install Siebel.

    c. **mkdir Siebel**

       Creates a Siebel folder. You use this folder as a base for the entire Siebel CRM installation, keeping everything in one place.

    d. **mkdir Siebel\Keys**

       Creates a folder under the /Siebel folder in which you will create the keystores.

    e. **cd Siebel\Keys**

       Changes the current directory to the newly created Siebel\Keys folder.

    f. **keytool -genkey -alias siebel -keystore .\siebelkeystore.jks -keyalg RSA -sigalg SHA256withRSA -keysize 2048 -validity 1825 -dname "cn=myserver.company.com"**Creates a keystore for the server: myserver.mycompany.com

       You must replace all occurrences of this with the actual server name. The validity attribute key is valid for 5 years (1825 days), you can change this value as appropriate.

       > **Note:** You will be prompted for two passwords — one for the key and one for the keystore. Due to downstream dependencies, these passwords must be the same. Note these passwords carefully, as they are required several times during the installation process.

    g. **keytool -list -v -keystore .\siebelkeystore.jks**Verifies that the keystore was created and checks for obvious errors.

    h. **keytool -certreq -alias siebel -keystore .\siebelkeystore.jks -file .\certreq.csr** Generates a certificate signing request (CSR). This will be provided to your CA to get an actual certificate.

About Selecting a Certificate Authority

The following options are available when choosing a certificate authority (CA):

- Trusted Root Certificate Authority. These are external companies who are inherently trusted by various operating systems and browsers, such as Symantec or GeoTrust. This is the preferred option because most components, including end-user browsers, will trust a certificate issued by a Trusted Root CA without requiring special action. This is a required approach for any environment that will be Internet-facing, such as a customer or partner portal.
- For environments that are specifically internal, you can use an existing internal CA to issue a certificate. In this scenario, machines within the corporate network will typically already be configured to trust the internal CA,

and therefore a certificate signed by that CA to servers running Siebel CRM will inherently be trusted by internal clients.

> **Note:** You cannot use this method for Internet-facing environments. External computers and other devices will not trust the certificate and will therefore throw security warnings when a user accesses the Siebel CRM environment. This method is sufficient only for specifically internal production environments and development, such as test or training environments and so on.

- When no internal CA is available and it is not feasible to obtain a certificate from an external Trusted Root Certificate Authority, you can create your own CA. However, anyone who accesses this environment will receive certificate security errors unless they manually import the root certificate for your CA. We recommend that you consider this method only when no other options are possible, and that it is used only for development and test environments.

After selecting a CA, submit your certificate request and the CA will issue an actual certificate.

The CA provides the specific information required to make the certificate request. An important consideration during this process is how you select Subject Alternative Names (SANs). These allow multiple machines to use the same SSL certificate. For example, if you want to install Siebel CRM components on five servers: server1.mycompany.com, server2.mycompany.com, server3.mycompany.com, and so on, you can create a single certificate request on any machine, providing all five machine names to the CA as SANs. You can then install the resulting certificate on all of your servers. It is also possible to request a wildcard certificate, for example, *.mycompany.com, which can be used on all servers.

Importing the Certificate Request

After you generate the certificate request, you submit it to your chosen certificate authority (CA). You then import the returned certificate into the keystore. Because your servers rely completely on your keystore, you need to import the root and any intermediate certificates. These are provided by the CA and you must import them to the same keystore where you created the certificate request.

1. Open a command prompt window.

2. Execute the following commands:

    a. **keytool -import -alias my_certificate_authority -file rootcert.cer -keystore .\siebelkeystore.jks**

       Imports the root CA certificate into the keystore. You are prompted to confirm if you trust the certificate.

    b. **(Optional) keytool -import -alias intermediate_certificate_authority -file intercert.cer -keystore .\siebelkeystore.jks**

       Imports an intermediate CA certificate into the keystore if required. Your CA will inform you if this is a requirement.

    c. **keytool -import -alias siebel -file my_cert.cer -keystore .\siebelkeystore.jks**

       Imports the Siebel certificate into the keystore.

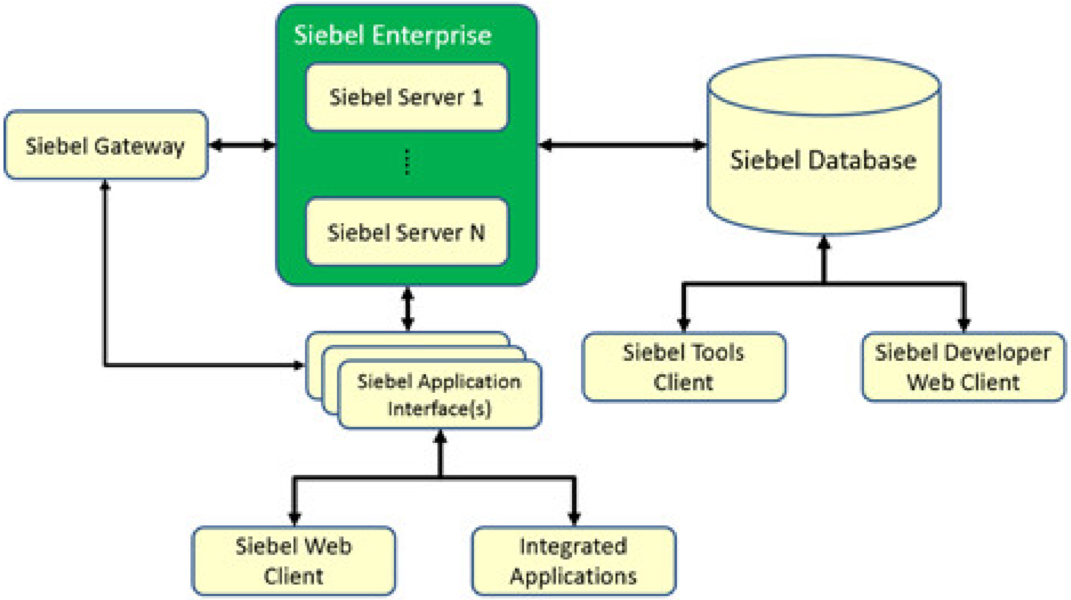
    d. **keytool -import -alias my_certificate_authority -file rootcert.cer -keystore .\TrustStore.jks**

       Creates the truststore and imports the CA certificate. This is the first reference to the truststore, so it prompts you for a password in order to create it and then prompts you as to whether or not to trust the certificate.

       > **Note:** For simplicity, it is recommended that you use the same password for both the Siebel Keystore and the TrustStore, as the installation wizards for Siebel CRM request both the stores, but only ask for one password. If your keystore and truststore have different passwords, you will need to manually edit the server.xml file after completing the installation. This file is located in the applicationcontainer\conf folder for that component.

Installing Software for the Siebel CRM Application

There are a number of software components required to implement Siebel CRM fully, depending on your requirements and your implementation. The following image shows the minimum required components and how they interact with each other.



As shown in this image, there are three distinct software installers that you will use to create your development environment:

- **Siebel Enterprise Software.** This provides the required software for the following components:
  - **Siebel Gateway.** This uses the embedded application container functionality to manage communications with other modules. Configuration data for the Siebel environment is stored internally to the Siebel Gateway.
  - **Siebel Enterprise.** This is a logical profile that defines the physical storage locations used by the Siebel Servers. For example, database type, name, location, and credentials, as well as the Siebel File System location and other global parameters.
  - **Siebel Server.** This is the application server that provides the interface to the Siebel database and file system, enforcing business logic. This is typically a load-balanced component running parallel on multiple servers.
  - **Siebel Application Interface.** This provides the delivery of Web pages from the Siebel Server to Web clients.
  - **Siebel Database.** The Siebel CRM product stores all of its configuration information and user data within a customer-selected database, for example Oracle, Microsoft SQL Server, or IBM DB2. The Siebel Enterprise Software installer provides the scripts necessary to populate this database with the required tables, indexes, data, and other objects necessary to run Siebel CRM.
- **Siebel Tools.** This is a 32-bit Windows application that allows developers and administrators to make changes to the Siebel repository. You can implement many changes using the Siebel Web Client, however in Siebel Innovation Pack 2017, there are some specific types of repository changes that require the 32-bit Windows Siebel Tools client, such as modifications to Workflow processes.
- **Siebel Developer Web Client.** This is a 32-bit Windows application that allows developers and administrators to interact with the Siebel Database without needing any Siebel Server components. While it is not mandatory to run the application, having Siebel Developer Web Client available typically facilitates certain configuration and deployment activities, such as debugging scripts or workflow processes.

During a basic installation, for example in a development environment, you only require a single instance of each component, but you can chose to install additional servers. For a test or production environment, it is likely that there will be more than one Siebel Enterprise Server and more than one Siebel Application Interface. This is to provide load balancing and resiliency.

This guide describes a minimal configuration, installing one of each component. For more information, see *Siebel Applications Administration Guide* .

Installing the Siebel Enterprise Software

To install the Siebel Enterprise Software, do the following:

1. Run the setup.bat or setup.sh file from the Siebel_Enterprise_Software\Disk1\install folder created by Siebel Network Image Creator (SNIC).

   This launches an installation wizard in which you can accept many default settings. Some of the critical installation prompts and recommended responses are as follows:
   - **Application Install Task.** If you are performing a new installation of Siebel, then select New Installation. Otherwise, select Migrate Installation. This guide focuses on new installations. For information pertaining to all types of installation, see *Siebel Installation Guide for Microsoft Windows* or *Siebel Installation Guide for UNIX* as appropriate.
   - **Installation Directory.** You can install all Siebel components under a single parent directory. To simplify installation, install everything under x:\Siebel, where x is a local drive, such as C: or D:. If installing Siebel CRM on UNIX, you can install in a similar location on your UNIX machine. In this guide, we install all

components to C:\Siebel. The installer adds the ses folder automatically as a subfolder under Siebel. The final path is C:\Siebel\ses

- **Components.** In a non-production instance, it is typical to install the following components together:
    - Gateway
    - Siebel Server
    - Database Configuration Utilities

  When these components are selected, the installer issues a warning message indicating that you cannot install these components together in an end-user environment. You are installing for a development instance, therefore you can ignore this warning.
- **Languages.** At a minimum, it is recommended that you select English (ENU), even if it is not your intended primary language.

  > **Note:** It is not possible to add languages after the initial installation. If there is any possibility that you require more languages in Siebel CRM, install all potential languages during your initial installation. Other than disk space (approximately 500MB per language), there is no negative impact to installing additional languages.

- **Port Configuration.** In a development or test environment, it is very possible that all components will be on the same physical server, as is described in this guide. For this reason, the default settings, which use standard HTTP and HTTPS ports for the application containers, are inappropriate. We need to reserve port 443, the HTTPS port, for traffic between end-user browsers and the Siebel Application Interface. The following port settings are therefore recommended for the Siebel Enterprise Server components:
    - **HTTPS Redirect Port.** 9010
    - **HTTP Connection Port.** 9011
    - **Shutdown Port.** 9012
- **Client Authentication Parameters.** These parameters indicate the Secure Sockets Layer (SSL) certificate keystores you need to copy into the installation location for the Siebel Application Interface:
    - **Keystore Name.** The location of the keystore, in this example, C:\Siebel\Keys\siebelkeystore.jks
    - **Truststore Name.** The location of the trust keystore for your certificates, in this example C:\Siebel \Keys\truststore.jks
    - **Password.** The password you used when you created your keystores.

      > **Note:** If you did not use the same password when creating the keystores, then you must edit the file C:\Siebel\ses\applicationcontainer\conf\server.xml manually after installation to set the passwords correctly.

    - **Gateway TLS Port.** This can be any available port, except in a migration installation from an earlier release of Siebel CRM, in which case, it must match the port for the previous Gateway installation. For simplicity, we will continue the numbering sequence and use 9013 for this port.

2. To facilitate re-installation, or additional automated installation, you are prompted to save your responses when you have completed each installation wizard.

Installing the Siebel Application Interface

The Siebel Application Interface (SAI) replaces the Siebel Web Server Extension (SWSE or SWE) used in previous versions. It provides a standards-based HTTPS entry point to the Siebel CRM application and is installed using the same Siebel Enterprise Server installation media as the Gateway, Siebel Servers, and Database Configuration Utilities.

However, although it may be installed on the same physical server as these components, it must be installed separately and into a different directory structure from the components installed previously.

To install the Siebel Application Interface, do the following:

1. Re-run the Siebel Enterprise Software installer.

   In many cases, the options for installation will be the same as those used when installing the Gateway, Siebel Servers, and Database Configuration Utilities. You must use the following installation prompts and recommended responses for Siebel Application Interface, described as follows:

   - **Installation Directory.** This must be different from the previous directory used for the Siebel Enterprise Server. The recommended path is C:\Siebel\sai
   - **Components.** Only install Siebel Application Interface.
   - **Languages.** You must install the same set of languages selected during the previous installation. Other than disk space, there is no negative impact to installing additional languages.
   - **Port Configuration.** The HTTP and HTTPS ports will serve as the entry point into the application (with the HTTP port automatically redirecting to the HTTPS port). If you do not want users to have to enter a port, set these to ports 443 and 80 (the default ports for HTTPS and HTTP respectively). The Shutdown port must be unique. We will continue numbering from the Siebel Enterprise Server installation and use 9014. The following port settings are therefore recommended for the Siebel Application Interface components:
     - **HTTPS Redirect Port.** 443
     - **HTTP Connection Port.** 80
     - **Shutdown Port.** 9014
   - **Username and Password.** The credentials that you specify will be used for accessing the Siebel Management Console initially, which is the user interface that we will use to configure the Gateway, Enterprise, Siebel Servers, and other components. After the initial setup, that is, once you have configured the Gateway Security Profile, you will use the username and password for the configured authentication authority. Any values are acceptable; for simplicity use: sadmin.
   - **Client Authentication Parameters.** These indicate the SSL certificate key stores (that we created earlier) which will be copied into the installation location for the Siebel Application Interface:
     - **Keystore Name.** The location of the keystore we created previously.

       For example: C:\Siebel\Keys\siebelkeystore.jks
     - **Truststore Name.**The location of the trust keystore for our certificates.

       For example: C:\Siebel\Keys\truststore.jks
     - **Password.** Whatever password you used when you created your keystores.

       > **Note:** If you did not use the same password when creating the keystores, you will have to manually edit the file C:\Siebel\ses\applicationcontainer\conf\server.xml after installation to set the passwords correctly.

About Installing the Latest Monthly Update

Before proceeding to the configuration steps, it is very important that you install the latest Monthly Update. While it is possible to install Updates later, it is possible that an Update could contain bug fixes, repository data, or seed data that would not otherwise be included in your Siebel database. Only the most recent Monthly Update is required for installation, as they are cumulative. For example, if the 18.5 Update is available, it can be installed directly on top of the 17.0 Innovation Pack without the intermediate installation of 17.1, 17.2, 18.4, and so on. To install the Monthly Update, run the installers on the same machines on which you previously ran the base Innovation Pack installers.

> **Note:** If a given base Innovation Pack installer was run more than once, then you must run the Update installers a corresponding number of times. For example, if you run the Siebel Enterprise Server installer to install the Gateway, Siebel Server, and Database, and then run it once more to install the Siebel Application Interface on the same server, then you must run the Update installer twice, directing the installer to each of the ORACLE_HOMEs specified during the installation of the Innovation Pack.

Verifying the Siebel Application Interface Installation

You can quickly verify that your Siebel Application Interface (SAI) installation was successful.

1. Browse to https://myserver.company.com/siebel/smc

   If your browser displays a security warning, this is because your browser does not trust your SSL certificate or the certificate authority (CA) that you used to sign it.
2. Enter your credentials in the login window.

   These are the credentials you specified when you installed the SAI component.
3. You are prompted to create a gateway — for the purposes of this test, you can ignore this message as we are just verifying the installation.
4. Close the browser.

Automatically Starting SES and SAI Components

Although the Siebel Enterprise Server and Siebel Application Interface components will start automatically after installation, they do not run automatically after a restart of the server. Before proceeding further, it is recommended that these be configured to start automatically. The specific method for doing this will vary by operating system. For example, on Windows, you could add the scripts to the Local Computer Startup Scripts Policy so that they will be run automatically every time the computer is restarted, while on UNIX, a shell script could be placed in /etc/init.d/.

The script to startup each component is named startup.bat (or startup.sh for UNIX) and is found in the ...\applicationcontainer\bin folder for each component, for example C:\Siebel\sai\applicationcontainer\bin\startup.bat.

After configuring the scripts to run automatically, restart the machine(s) before continuing to validate that automatic startup process is working, and then proceed to the configuration of your Siebel Database.

Creating the Siebel Database

Before creating the Siebel Database, review the following topics:

- *About Installing Client Connectivity Software*
- *Prerequisites for Installing the Database*

Then do the following:

- *Running the grantusr.sql Script*
- *Installing the Siebel Database*

About Installing Client Connectivity Software

Before you create the Siebel database, you must ensure that you have the appropriate client connectivity software.

This varies by database platform:

- **Oracle Database.** You must install the 32-bit Oracle client, even if you are running on a 64-bit machine. This is available for download from Oracle Technology Network. Consider the following guidelines when installing Oracle Database:
  - When installing the software, select the Administrator option. Do not select the Instant Client or Runtime options.
  - Before installing the client on a Windows server, re-install the Microsoft Visual C++ 2010 Redistributable Package (x86). This prevents errors during the client installation process. Ensure that you run the installer as a local administrator.
  - If installing on a Windows server that also has a 64-bit version of Oracle installed, you need to edit the following registry value before running the installation: HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Oracle\inst_loc, where the Value name field is inst_loc and the Value data field is c:\\Program Files (x86)\\Oracle\\inventory
- **Microsoft SQL Server.** Install the Microsoft SQL Server Native Client, available from Microsoft.
- **DB2.** Install the IBM DB2 Connect 32-bit client (even if running on a 64–bit operating system). Select from the versions noted on the Certifications tab on My Oracle Support that are compatible with your DB2 database server version according to IBM's certification matrix.

Prerequisites to Installing the Database

Before we can start installing the Siebel Database, the database administrator must perform some preliminary steps. These are documented in detail in the *Siebel Installation Guide for Microsoft Windows* and *Siebel Installation Guide for UNIX* , but here is a short description:

- **Oracle.** Create a tablespace in which the Siebel database objects will live. The recommended name is SIEBEL.
- **Microsoft SQL Server.** Create a database. The recommended name is siebeldb.
- **DB2.** Create the SIEBEL and other database users.

> **Note:** These steps are typically done by a database administrator who may or may not be a direct member of the Siebel development team.

Running the grantuser.sql Script

A database administrator, who may or may not be a direct member of the Siebel development team, performs these steps.

After preparing the database, run the grantusr.sql script found in the .../dbsrvr/{db_platform} folder (for example, C:\Siebel\ses\dbsrvr\ORACLE). There are brief instructions at the beginning of each script explaining how to run the script. It will prompt you for information about the preparatory work done in the previous step, for example. the server where the database resides, the name of the database, any tablespaces, usernames, passwords, and so on. When the script completes, you will have an empty shell of a Siebel database with all the required users. Note that the execution of the script is typically done by a database administrator who may or may not be a direct member of the Siebel development team. This allows the Siebel team to have everything they need to populate the Siebel CRM Database without the need for specialized database administrator skills. The Siebel team should provide the desired passwords for the database users that the script will create. These include:

- **SADMIN.** The Siebel Administrator user will be used for administering the Siebel CRM application throughout the implementation.
- **GUESTCST.** The anonymous Guest Customer user. This user's credentials are used automatically by customer-facing applications before a user has actually logged in, providing access to the login page itself. This allows access to non-sensitive areas of the product, such as Branch Locators, without the need for every visitor to have a login.
- **GUESTERM.** The anonymous Employee user. This user's credentials are used automatically by employee-facing applications before a user has actually logged in.
- **LDAPUSER.** This is a generic user used by the authentication system.

After the script has been executed, test the SADMIN login using a database platform-specific tool, such as SQL*Plus for Oracle, to verify that the user was correctly created.

Installing the Siebel Database

The Siebel CRM database is installed using the Database Configuration Wizard. You can execute it using the Database Server Configuration shortcut from the Windows Start menu or from the following UNIX shell: $SIEBEL_ROOT/config/config - mode dbsrvr

The configuration wizard prompts you for information on the location of the database, database names, tableowners, tablespaces, and so on, depending on your database platform. For more information, see *Siebel Installation Guide for Microsoft Windows* or *Siebel Installation Guide for UNIX* as appropriate. The following are important points to consider during database installation:

- When referring to directory locations, specify the path where the Siebel Enterprise Server (SES) components are installed, not the Siebel Application Interface (SAI) components.
- When performing the Siebel User/Role Creation step, ensure to change the default selection to: “GRANTUSR.SQL has been run by the DBA to create Siebel users and roles”.

  This is because you already executed the grantusr.sql script. If you do not update this selection, the installation will not continue.
- Ensure that the Database Encoding value is Unicode. Oracle does not support non-Unicode Siebel CRM databases for new installations.
- If you are installing one and only one language, you may pick it here. However, if there is any possibility that you will ever be installing multiple languages, it is recommended that you select English as the base language. This will default all log files and other settings to English, making it easier to debug when working with Oracle Technical Support. You will be able to install additional languages later.

The Configuration Wizard displays all the information you entered, and allows you to apply the changes. Once you confirm these changes, the Siebel Upgrade Wizard displays the steps necessary to install the database. Click OK to start the Siebel database installation. When the installation is complete, review the log files before continuing. For more information, see the installation guide for your selected database.

Configuring the Siebel Gateway

The Siebel Gateway maintains the configuration information for the components in the Siebel CRM environment. For example, it monitors where each Siebel Enterprise Server (SES) and Siebel Application Interface (SAI) instance is running, which components should be run on which physical servers, and so on.

Do the following to configure the gateway:

1. Navigate to the Siebel Management Console (SMC) at https://myserver.company.com/siebel/smc.

   This is the server where you installed the SAI components. If you configured a HTTPS port number different from the default value, then include this in the URL.
2. Enter the credentials that you defined when installing the SAI.

   The SMC automatically determines that a Siebel Gateway has not yet been configured and starts the Gateway configuration process.
3. Enter the requested information.

   For more information, see *Siebel Installation Guide for Microsoft Windows* or *Siebel Installation Guide for UNIX* as appropriate. The following are key attributes:
   - **Host Name:** HTTPS Port

     - **Host Name.** This is the host name where the SES components are installed and where you intend to run the Gateway. This host name must match one of the Subject Alternative Names in the Secure Sockets Layer (SSL) certificate previously generated and signed.
     - **HTTPS Port.** This is the HTTPS port you selected when you installed the SES components, not SAI. For example: myserver.mycompany.com:9010
   - **Security Profile.** This sets up the security mechanism used to authenticate connections to SMC. For the Gateway, you can name it, for example, Gateway Security Profile. For a development instance, you typically use database authentication, so you must specify the database type, for example, Oracle or Microsoft SQL Server. You also must specify the connection information, such as the location of database server, port for TNS Listener, and so on.
   - **Testing.** Enter the credentials for SADMIN. These are the same credentials as those used when executing the grantusr.sql script. This password may be different than the one used during the installation of SAI.
4. If the test is successful, then SMC logs out, and prompts for login credentials. At this point, use the SADMIN credentials defined by the grantusr.sql script, and not those used during the installation of SAI.
5. The configuration wizard prompts you for service user credentials. These are the credentials under which the Gateway Service runs (either as a Windows Service or as a UNIX process).

   > **Note:** For Windows-based systems, this should be a domain user who is also a local administrator.

When you complete these steps, the Siebel Gateway service will be created, and the Siebel Deployment Hierarchy page opens.

Configuring the Siebel Enterprise

The Siebel Enterprise maintains environment-wide settings that are shared by all components, such as the location of the Siebel File System, the database, and so on.

Before continuing, it is important to understand that in the Siebel Management Console (SMC), a profile represents a template that provides the information needed for a given component. However, a profile is not an actual instance of that component. Using profiles allows us to deploy many instances of the same entity quickly, and repeatedly. You deploy the Enterprise only once, whereas a Siebel Server or Siebel Application Interface profile can be used many times.

Creating the Enterprise Profile

To create the Siebel Enterprise profile, navigate the Siebel Management Console (SMC) to the Profiles/Enterprise node and create a new profile. For more information on each parameter, see *Siebel Installation Guide for Microsoft Windows* or *Siebel Installation Guide for UNIX* as appropriate.

Some of the key parameters are as follows:

- **Profile.** A profile is a template for a component that can be deployed many times. To distinguish a profile from an instance of a component, it is recommended that profile name always includes the word: profile. For example, the Enterprise profile could be named EnterpriseProfile.
- **Primary Siebel File System.** The Siebel File System will be used to store documents associated with Siebel CRM data, such as spreadsheets, word processing documents, presentations, images, and so on. This must be a network location that allows read-write access to all the servers on which Siebel Servers are eventually deployed. While the Enterprise creation requires only a single Siebel File System, it is possible to create multiple file systems to improve load-balancing and performance. For more information, see *Siebel System Administration Guide* in Siebel Bookshelf.
- **Authentication.** The username and password for the Siebel Gateway.
- **Security Encryption Level.** Setting this to SISNAPI Using TLS 1.2 will cause inter-process communication amongst Siebel Enterprise Components to use TLS, improving security for these conversations. If this option is selected, it will be necessary to provide information about the Certificate Authority and other information needed to encrypt those conversations.

Deploying the Enterprise Profile

After successfully creating the Enterprise profile, deploy it as follows:

1. Navigate the Siebel Management Console (SMC) to the Siebel Deployment Hierarchy and click the plus (+) icon to add an Enterprise to the deployment.
2. Select the profile you created.
3. Assign a name to the Enterprise.

   A profile is a template, and the name provided here is only a named instance of that profile. The name of the Enterprise can be any alphanumeric string, but should be meaningful and easily identifiable, such as Development, or Test.
4. Select Deploy.

   If you select Stage, then nothing will be deployed.

When you submit the request, the SMC does the following:

- Creates the Siebel File System.
- Creates an ODBC source named {EnterpriseName}_DSN. This is used to communicate with the database, for example, Siebel_DSN.

5. Refresh the Siebel Deployment Hierarchy page until the Enterprise shows as fully deployed.
6. Verify that the File System and ODBC source were created:
   - **Siebel File System.** You can verify that this was successful by inspecting the folder specified during the creation of the Profile. The Profile will show a set of folders that will be eventually used by the Siebel File System Manager.
   - **ODBC Source.** For Windows operating systems, validate that the ODBC source was created under the 32-bit ODBC Control Panel. For UNIX, check the contents of the .odbc.ini file in:

     /SIEBEL_HOME/ses/siebsrvr/sys

     For example:

     ```
     [ENT1_DSN]
     Driver=/siebel/ses/siebsrvr/lib/SEor827.so
     ColumnSizeAsCharacter=1
     ColumnsAsChar=1
     ArraySize=160000
     ServerName=SIEBELDB
     ```

Configuring Siebel Servers

Siebel Servers are the application servers that provide the business logic layer for Siebel CRM. A Siebel Enterprise must include at least one Siebel Server, but in most cases, particularly for non-development instances, there is more than one Siebel Server.

You would deploy more than one Siebel Server for one of the following reasons:

- **Performance and Load Balancing.** Siebel CRM is very scalable horizontally. Additional Siebel Servers allow more users to connect to the system, allow the execution of more background jobs, and so on.
- **Specialization.** In addition to providing the business logic layer for active users, Siebel Servers run background jobs, such as large data import jobs or integration with other systems. To avoid these jobs interfering with the user experience, it is common to deploy more than one Siebel Server specifically for this reason.
- **Operating System Dependence.** Siebel CRM software runs equally well on Microsoft Windows and the various UNIX platforms on which it is certified. However, in some instances Microsoft Windows is specifically required. For example, Document Generation which interacts with Microsoft Word via Microsoft's Component Object Model (COM) technology and Siebel Server Sync for Microsoft Exchange (SSSE), which relies on Windows Integrated Authentication to interact with Microsoft Exchange. It is not uncommon for a customer to run most of their Siebel Servers on a UNIX platform, and to use a few Microsoft Windows servers to support these specific use cases.

Creating the Siebel Server Profile

As with all Profiles, the Siebel Server Profile is created in SMC: in this case, under the Profiles area, Siebel Servers node. Unlike the Enterprise Profile, it is possible, and in fact likely in a non-development environment, that you will create multiple Siebel Server profiles. The following example profiles illustrate possible examples how you might do this:

- **SiebSrvrUIProfile.** Supports end-user User Interface sessions.
- **SiebSrvrWFProfile.** Supports background workflow processes.
- **SiebSrvrEIMProfile.** Supports Enterprise Integration Manager processes for bulk data loads.
- **SiebSrvrRESTProfile.** Supports REST integration.

Different profiles allow you to define specifically which application services are available on a given Siebel Server and, if needed, easily deploy an additional instance of that set of services onto additional servers for high availability and scalability.

When determining how many profiles to create, you must consider the different possible ways that users and other systems interact with Siebel CRM. You can add more profiles and Siebel Servers at any time, and you can deploy and redeploy a Siebel Server quickly if needed.

For assistance determining number of profiles, servers and so on based on, customer requirements and expected usage, contact the Oracle Advanced Customer Support team (Expert Services).

Create the Siebel Server profile using the SMC as follows:

1. From the Profiles/Siebel Servers node in SMC, create a new Profile record.
2. Select the Siebel Server Components that particular profile will support.
    - At a minimum, you must create a profile which includes the following sections:
        - Siebel Web Tools. This is used for making configuration changes to the application.
        - Each Component Group that includes the application that will be used. For example, if you plan to use Call Center, select the Siebel Call Center Component Group, if you plan to use Financial Services, select the Siebel Financial Services Component Group, and so on.
        - Enterprise Application Integration (EAI). It is required to select this if you need access to the REST API.
    - You should also include any component group that contains a component that is used for the specific implementation, such as Siebel eDocuments, which is used for document generation on Windows servers only, and so on.

    > **Note:** A Component Group typically contains many individual server components. For example, the Enterprise Application Interface Component Group contains more than ten server components, each of which automatically starts when the Siebel Server service starts. It is likely that only one or two of them is required. To conserve resources on the server, the unwanted components can be disabled using the Siebel Server Management utility. For more information, see *Siebel System Administration Guide* in Siebel Bookshelf.

3. Configure the Advanced Settings tab for load balancing and high availability.

    These are typically not required for development environments, but can be used for other environments. For more information, see *Siebel Installation Guide for Microsoft Windows* or *Siebel Installation Guide for UNIX* as appropriate.

Deploying the Siebel Server Profile

After successfully creating the Siebel Server profile, deploy it by doing the following:

1. In the Siebel Management Console (SMC), navigate to the Siebel Deployment Hierarchy and click the plus (+) icon.
2. Select Siebel Server.
3. Enter the following information:
    - **Host and Port.** This the host where the Siebel Enterprise Servers (SES) components are installed and the HTTPS port number that was specified during that installation (for example, myserver.mycompany.com:9010).
    - **Action.** If you are planning to deploy immediately, select Deploy.
    - **Siebel Server Name.** It is recommended that you use the name of the physical host. For example, if your host is myserver.mycompany.com, then use myserver as the host. On rare occasions, you might deploy more than one Siebel Server per machine, for example, where Siebel Servers are clustered in Active-Active configuration for high availability. In this scenario, add an identifier to the server name to distinguish each from the other, for example, myserver_01, myserver_02, and so on.
    - **Languages.** These define all the languages this server supports. This selection is applicable to all Object Manager components, for example, user interface applications such as Call Center or Financial Services, and EAI integration components. However, this does not apply to components that are language-independent, such as Enterprise Integration Manager.

    When you submit the request, SMC does the following:

    - Creates a Window Service or a UNIX daemon to host the selected components.
    - Launches Siebel CRM processes to host these components. You can view these processes in Windows Task Manager or the output of the ps utility on UNIX. All Siebel processes will match the pattern: sieb*.
4. Refresh the Siebel Deployment Hierarchy page until the Siebel Server shows as fully deployed.

    This might take several minutes. If deploying additional Siebel Servers to other machines using the same or different profile, you can deploy these profiles in parallel without waiting for earlier deployment(s) to complete.

Configuring the Siebel Application Interface

The Siebel Application Interface (SAI) provides the external interface to the Siebel CRM application. This includes user interface channels, such as employees, customers, and partners using browsers to interact with the application, as well as integration with other applications through REST or Web service calls. Normally a development instance requires a single SAI, but as with the Siebel Server, the Application Interface can be enabled on multiple servers to provide load balancing and high availability. For more information, see *Siebel Installation Guide for Microsoft Windows* or *Siebel Installation Guide for UNIX* as appropriate.

The key parameters of the SAI are those that are affect client interactivity, such as session timeouts.

Consistent with the Enterprise and Siebel Servers, the SAI, configuration is completed in two parts:

- Creating or more SAI profiles. For many environments, it is likely that you only need one a single profile, even if you will have Application Interfaces running on multiple machines. You might consider creating multiple profiles for one of the following reasons:
  - **Differing authentication.** For example, using a different authentication authority for your internal users from that which you will use for your Internet users.
  - **Isolation.** For example, to place Application Interfaces supporting customer-facing applications on one set of servers and employee-facing applications on another.
- Deploying the SAI profile as many times as required to support capacity requirements, high availability, isolation, and other technical and functional requirements.

Creating the Siebel Application Interface

You create the Siebel Application Interface (SAI) profile in the Siebel Management Console (SMC), under the Profiles area/Application Interfaces node. It is recommended that the profile name contains the word: profile. For example, SiebelAppIntProfile.

At least one Siebel Server with an EAI Object Manager must be deployed before you can create the first SAI profile. This is because the SAI profile requires selection of an existing EAI Object Manager.

Create the SAI profile using the SMC as follows:

1. From the Profiles/Applications Interface node in SMC, create a new Profile record.
2. Provide the requested information.

   The following are important points to consider:

   - **Basic Information Tab:** Enter the following information.
     - **HTTP 1.1-Compliant Firewall / Enable Web Compression.** Set this to TRUE, unless there is a firewall that prevents HTTP 1.1 traffic.
     - **Configure Fully Qualified Domain Name.** If you have a single machine hosting an Application Interface, or are using a single entry point, for example a load balancer, set this parameter to TRUE. You must also specify the fully-qualified domain name, for example, myserver.company.com.
     - **Authentication Information.** Enter the relevant information for your environment.
     - **Logging.** You can accept the default values. However, if you are debugging a problem, you can enter different values.
     - **REST Inbound Defaults.** Select an EAI Object Manager, for example EAI Object Manager (ENU), and specify a base URL. This URL must include the name of a server where the

Application Interface is running and includes the HTTPS port. For example, https://myserver.company.com:443/siebel/v1.0

- **Applications Tab:** Define the entry points that users use for connecting to Siebel CRM through the browser and that other applications use for integration. For each application that will be exposed, click the plus (+) icon and provide the following key information:
    - **Application Name.** This is the entry point that the user uses to navigate to Siebel CRM. For example, if the application name is callcenter, the user navigates to https://myserver.mycompany.com/siebel/app/callcenter/enu to reach the application.
    - **Object Manager.** Select the object manager that maps to the Siebel CRM application, as defined in the runtime repository that will be mounted at this entry point. For example, for the Call Center application, select Call Center Object Manager (ENU).
    - **Language.** Select the language that you want to expose. To expose a language, you need to have selected this language during the installation of both the Siebel Enterprise Server (SES) and SAI components, and also you need to have imported the language using the Database Configuration Wizard.
- **REST Inbound Resources.** For many environments, this field does not require configuration. For more information, see *Siebel REST API Guide* .
- **Other Information.** Accept the default parameters, unless you encounter specific instructions documented in Siebel Bookshelf to support your business requirements.

Deploying the Siebel Application Interface

After successfully creating the Siebel Application Interface (SAI) profile, deploy the profile by doing the following:

1. Navigate the Siebel Management Console (SMC) to the Siebel Deployment Hierarchy and click the plus (+) icon.
2. Select Application Interface Server.
3. Enter the following information:
    - **Host and Port.** This the host where the SAI components are installed and the HTTPS port number that was specified during that installation (for example, myserver.mycompany.com:443).
    - **Action.** If you are planning to deploy immediately, select Deploy.
    - **Application Interface Node.** It is recommended that you use the name of the physical host. For example, if your host is myserver.mycompany.com, then use myserver as the host.

    When you submit the request, the SMC instructs the SAI to expose an entry point into the Siebel CRM application for each application defined in the profile. For example: https://myserver.mycompany.com:443/siebel/app/callcenter/enu and https://myserver.mycompany.com:443/siebel/app/webtools/enu.

    Unlike other component types, you can make changes to the SAI profile that have already been deployed, and these changes take effect immediately without needing to redeploy.

Adding the Licence Keys

Siebel CRM requires you to activate valid license keys in the database. Without valid license keys, any attempt to connect causes the following browser error: The server you are trying to access is either busy or experiencing difficulties. Please close the Web browser, open a new browser window, and try logging in again.

```
The Object Manager log file in the siebsrvr\log folder provides a more informative message for
 the administrator. For example in the SCCObjMgr_enu_XXXX_YYYYYYYY.log file:ObjMgrLicense Error 1
```

```
000000035a0b2200:0 2YYYY-MM-DD HH:MM:SS (dmlicmgr.cpp (308)) SBL-DAT-00172: No valid license keys were
found in the database.
```

To add the license key, do the following:

1. On any server where the SES components are installed, navigate to the Siebel Server ..\ses\siebsrvr\bin folder.
2. Run the script licensekeymodule.bat (Windows) or licensekeymodule.sh (UNIX) script.
3. Enter the following connection information:
    - **Siebel Server Location.** The root of the siebsrvr folder. For example: C:\Siebel\ses\siebsrvr
    - **ODBC DSN.** The ODBC source created by the deployment of the Enterprise Profile, which will be named enterprise_DSN where enterprise is the name of the Enterprise defined by the Profile. In this example, it is named siebel_DSN.
    - **Tableowner / Username / Password.** The database credentials specified when the grantusr.sql script was executed. This is usually SIEBEL, SADMIN, and the SADMIN password, respectively.
    - **DB Platform.** Select one of the following: ORACLE, DB2UDB, DB2390, or MSSQL
    - **Log folder.** This can be any valid path. This is usually the Siebel Server log folder, for example: C:\Siebel\ses\siebsrvr\log.
4. Select all the licensed modules.

    > **Note:** Access to or possession of a license key, code, file, and so on, which unlocks or enables a software product is not a grant of entitlement. Your license agreement with Oracle provides the terms governing your use of Oracle's products. Please ensure you are using the products in accordance with your license agreement.

5. Click the Apply button and ensure that a success message is displayed.
6. Close the license key module.

    After adding license keys, users can immediately to log into the system.

    Ensure that the SADMIN user can log into https://myserver.mycompany.com:<HTTPS Redirect Port>/siebel/app/callcenter/enu, (or whichever entry point is defined in the Siebel Application Interface Profile).

Setting up the Siebel Tools Client

In Siebel Innovation Pack 2017, you perform most configuration changes, such as adding custom fields or validation rules, using Siebel Web Tools. However, there are some types of configuration that require the Siebel Tools client, such as development of Workflow processes or Task-Based UI, as well as certain functions associated with upgrades, such as reviewing conflicts, that are identified by the Incremental Repository Merge (IRM) process. For this reason, you need to install Siebel Tools on all machines that you use for these purposes for all Siebel CRM configuration team members.

Siebel Tools must have database connectivity software. Before proceeding, install the required software for your database platform on each machine where you install Siebel Tools. In Siebel Innovation Pack 2017, Siebel Tools also installs the Oracle Express (Oracle XE) database platform. As a result, all clients on which Siebel Tools is installed, require the Oracle 32-bit Client, even if your server database is Microsoft SQL Server or IBM DB2.

To install Siebel Tools, first install the base Innovation Pack release, for example 17.0, then the most recent Monthly Update (for example, 18.5). Monthly Updates are always cumulative. If there is more than one patch set available, you only need to install the latest version.

Unlike the Siebel Enterprise Server (SES) and Siebel Application Interface (SAI) components, installing and configuring Siebel Tools is done in a single step. During the installation, all the information that Siebel Tools requires is requested.

The majority of this information was specified during previous steps. For example, the server where the Gateway Server is deployed, the name of the Enterprise, and so on. You will be required to provide the following new information:

- **Directory Path for 32-bit Oracle Client.** The Siebel Innovation Pack 2017 Siebel Tools installer always installs the Oracle XE database, and therefore requires pre-installation of the Oracle 32-bit Database Client. Specify the path of the client here, for example, C:\Oracle\client\product\12.1.0\client_1.

> **Note:** When you install Siebel Tools, you are prompted to install the local database. Do not select and do not install the local database; Siebel Innovation Pack 2017 and later versions do not support the use of local databases for Siebel Tools.

After installing the base 17.0 release, perform the following tasks:

- Install the most recent Monthly Update.
- Verify that Siebel Tools connects to the Siebel CRM Enterprise database. You can do this by launching the Siebel Tools client, selecting the ServerDataSrc data source, and logging in using the SADMIN credentials.

Setting up the Developer Web Client

All administration activities can be performed using the browser-based user interface. However, it is sometimes convenient to have access to the Siebel CRM database using a Developer Web Client. This client does not rely on any of the Siebel Enterprise Server (SES) or Siebel Application Interface (SAI) components, but is a self-contained client with all the features of the browser client, with independent binaries.

This is convenient in the following situations:

- Troubleshooting server connectivity issues using the SAI.
- Logging in with a different authentication mechanism than is provided by the SAI. For example, logging in using database authentication when the SAI has been configured for a Single Sign-On (SSO) authentication mechanism.
- Debugging scripts, workflow processes, and Task-Based UI.

The Siebel Developer Client installation is similar to the Siebel Tools client installation; it is installed and configured in a single step. Much like Siebel Tools, it also requires the Oracle 32-bit client (even for customers using IBM DB2 or Microsoft SQL Server for their Siebel CRM Enterprise .database) and automatically installs an Oracle Express (Oracle XE) database as part of the installation process if you select the Sample Database option.

To install Siebel Developer Web Client, first install the base Innovation Pack release then the most recent Monthly Update (for example, 18.5). Monthly Updates are always cumulative. If there is more than one Update available, then you only need to install the latest.

The majority of the required information matches values specified during previous installation steps. You require the following new information:

- **Languages.** While Siebel Tools can be used to configure all languages in the Repository, you must install the Siebel Developer Web Client for all languages that you plan to deploy in your Enterprise.
- **Type of Client.** There are two types of clients available:
  - **Siebel Remote Client.** This is used by end users to work in an offline mode. For more information, see *Siebel Remote and Replication Manager Administration Guide* .
  - **Developer Web Client.** This is the recommended selection for administrative purposes.

- **Search Server Hostname and Port.** Accept the defaults provided.
- **HTTP Connection Port and Shutdown Port.** These are used for test automation and other purposes. If you install the Developer Web Client on a different machine from SES and SAI, it is still recommended that these ports be unique to avoid any possible confusion. The defaults of 9001 and 9005 are acceptable unless already in use.

When you finish installing the base release, ensure to perform the following tasks:

- Install the most recent Monthly Update.
- Verify that Siebel Developer client connects to the Siebel CRM Enterprise database. You can do this by launching one of the shortcuts it created, for example Siebel Call Center, selecting the ServerDataSrc data source, and logging in using the SADMIN credentials.

4 Basic Organization Setup

Basic Organization Setup

Once the Siebel CRM components have been installed and configured, it is tempting to immediately start implementing configuration and other changes to personalize the functionality for a particular set of business needs. However, spending a small amount of time doing some administrative work in advance of this step will set the stage for smoother development and testing cycles, eventually leading to a better user experience in a production environment. This chapter includes the following topics:

About Seed Data

When the Siebel CRM Database is created, it logically has the following components:

- **Physical Structure.** The physical structure includes tables, indexes, and a very small number of other database objects. Siebel CRM does not define business logic in database objects. For example, only nominal data validation (such as datatype) is done at the database layer. Siebel CRM does not use stored procedures or triggers in the base product except in rare cases. The physical database is 99% tables and indexes. This allows for the maximum amount of portability across the database platforms that Siebel CRM supports.

  > **Note:** There are some specific examples where triggers are added later in support of the workflow or integration, but the basic product does not include these at the time of installation.

- **Repository Data.** The Siebel CRM Runtime Repository contains metadata that defines how the Siebel CRM application behaves. For example, it defines which user interface objects a user can see, where to store data at the physical layer, validation rules, and workflow. Repository data is discussed in depth later in this guide and throughout the *Siebel Bookshelf* .
- **Seed Data.** This is data that is placed in the database to provide the user with basic options before any customization or configuration of the product has occurred. The following are a few examples of seed data:

    - **List of Values.** This type of seed data appears in Picklists throughout the Siebel CRM application. For example, consider the possible states for a Service Request record. The possible states can include: New, In Progress, Waiting on Customer, Working, Waiting on Engineering, or Closed. However, different options may apply to an Activity. The possible states for an Activity can include: Planned, Confirmed, In Progress, or Done. A List of Values contains a default set of options for Picklists. For more information about List of Values, see *About List of Values*.
    - **Currency.** Currencies facilitate interactions with the global economy. The Siebel CRM database is seeded with the known world currencies for users to select from and to assist with conversion between them. For more information about currency, see *About Currency*.
    - **Time Zones.** Time zones are a critical component in working with a global economy. Time zones support showing Siebel CRM interactions in the time zone of a given user, but are crucial in determining whether Service Level Agreements are being met. For more information about time zones, see *About Time Zones*.
    - **Responsibilities.** Responsibilities determine which application views a particular user can see. For more information about responsibilities, see *About Responsibilities*.

The various types of seed data have the following factors in common:

- Seed data is placed into the database at the time that the Siebel Database Configuration Wizard runs.
- Seed data can be inactivated or supplemented. For example, you may add a new Service Request status or remove an existing one.
    - Seed data should never be deleted. If it is not wanted or needed, it should be inactivated. The reason for this is that the Siebel Upgrade Process (Incremental Repository Merge, or IRM) will put it back the next time it is run.
    - Other than activation or inactivation, seed data should not be modified. For example, if you want to eliminate the Closed Service Request status and create a new one named Finished, the recommended approach is to inactivate the Closed record and create a new record for Finished, rather than change Closed directly to Finished.
        - This avoids issues during the Upgrade IRM process.
        - In some cases, specific seed data is required for the application to work properly. This is particularly true for objects that have specialized functionality, such as Calendar objects.
- Seed data may be modified in each release of Siebel CRM. For example, time zone rules change annually due to local laws. The Oracle Siebel CRM Engineering Team regularly update the time zone seed data to reflect current law.

> **Note:** In order to minimize modification of seed data in a non-recommended way, many seed data objects in Siebel CRM are intentionally locked down, allowing only minor changes, such as inactivation. A quick Internet search will provide customers with a way to work around this validation policy. It is strongly recommended that no customer take advantage of this opportunity and adhere to the validation rules implemented in the Siebel CRM application. Failure to respect this recommendation may result in a Siebel CRM application that cannot be supported by the Oracle Technical Support organization.

The remainder of this chapter focuses largely on some of the key seed data objects, as well as other basic administrative data used by the application, and how to maintain these objects. These include:

- *About Views*
- *About Responsibilities*
- *About Organizations*
- *About Positions*

- *About Users and Employees*

About Views

A *View* represents a page of information in the Siebel CRM application. A view can take many forms, including (but not limited to) the following:

- **Home Page Views.** Home Page Views are where a user is placed immediately after login. These views are typically made up of several regions (known as applets) that will help a user to see the most important information they need to decide where to start their day. For example, a Field Service user may see a list of open Service Requests and a Calendar showing where they have scheduled customer appointments, while a Sales user might see a list of their Sales Opportunities and outstanding Activities.
- **List Views.** List Views are where a user can see many objects of the same type. For example, a user may want to see a list of customer accounts, contacts, campaigns, or service requests. These views typically allow the user to drill into a particular item to a detail view in order to get more information about that object. For example, drilling into a specific customer account will provide access to related contacts, opportunities, service requests, or orders.
- **Detail Views.** Detail Views are where a user can see more detailed information about a given top-level (parent) object as already explained.
- **Specialized Views.** Specialized Views include charts, dashboards, or portals to other applications.

Views in Siebel CRM are defined in two places:

- **Runtime Repository.** The Runtime Repository stores the actual definition of the view. For example, what objects appear in the view, how are those objects laid out, what child objects does it include, and what high-level screen object is a part of the view.
- **Administrative Seed Data.** In addition to defining the view in the runtime repository, each view must be registered within the application. For more information about registering views, see *Registering Views*.

Another aspect to views is the scope of access that they provide. A given logical view may be defined many times with a different set of records displayed based on the user's access to records. As a simple example, consider a sales opportunity. The sales representatives working on that opportunity should see it, while representatives who are not involved should not. A first-line manager should be able to see all the sales opportunities that all of their direct reports are working on, while a second-level manager should see all the opportunities for their reports across all their intermediary managers. The following are the types of visibility rules available for views:

- **My Views.** My Views shows all of the objects to which you have a direct association. My Opportunities shows all sales opportunities for which a given user is directly involved. These are typical used by independent contributors.
- **My Team's Views.** My Team’s Views show all of the objects which meet one of the following criteria. The typical user is a manager at any level.
  - You are a manager and are personally involved. For example, you are working on a sales opportunity.
  - You are a manager and have a direct report involved. For example, an employee working for you is working on a sales opportunity.
  - You are a higher-level manager and have someone who directly or indirectly works for you involved. For example, you are a director with a manager under you who has a sales representative working on a sales opportunity.

- **All Views.** All Views show all of the objects across your particular Organization. This can be a little misleading, in that a large company may have many different Organizations in it. See the section on Organizations for more information, but as a simple example, consider a multi-national company. You may define an Organization as a region. For example, North America, Europe, or AsiaPac. You may want to be able to restrict visibility to users being able to see only those sales opportunities within their region. A typical user of these views is a senior executive or a cross-functional user, such as a technical support engineer who must support all customers.
- **All Across My Organizations.** All Across My Organizations view is similar to All Views. All Across My Organizations view allows a user associated with more than one Organization to see all the objects within all those Organizations at once. To continue the previous regional example, this would allow separate regions for Eastern Europe and Western Europe to be defined as Organizations and allow a user to be associated to both of those and see all sales opportunities in either region. Typical users are senior executives or cross-functional users, such as Support Engineers.
- **All Across All Organizations.** All Across All Organizations view shows all objects of a given type, without regard to owner or Organization. These views are typically reserved for very senior users, such as CEOs.
- **Administration.** Administration views are specialized views that allow access to all records of a given type and always allow editing of those records, even if a given field or even the entire record would normally be marked read-only. These are reserved for system administrators and are used to repair data, for example, cleanup mistakes and reassign data when it is not visible to those users who need it.

About Seed Data Views

Immediately after installing the Siebel CRM database, there are well over 10,000 Views in seed data. This is an illustration of how extensive the Siebel CRM application is and many use cases and business needs can be accommodated by one of these views without the need for additional development or customization.

Registering Views

In the event that a new view is required, making that view available to end-users requires a few separate steps. For information on the actual creation of Views, such as the layout or content, see *Configuring Siebel Business Applications*. Only after defining the view and releasing it to the runtime repository will you be concerned with registering the view for use.

After designing, configuring, and testing the new view, navigate to the Site Map, Administration - Application, and then Views. Create a new record using the same name as that used when defining the view in the Siebel Runtime Repository. The only option required is whether or not that view should be made available to Siebel Mobile users, as indicated by the Local DB checkbox. This concept is explained in the Siebel Bookshelf. As a simple rule of thumb, a view that could expose a lot of data, such as an All Opportunities View, should not be made available to remote users for performance reasons.

> **Note:** Registering the view in the Siebel CRM application makes it available to be assigned to users, but will not actually make it visible to them. For more information, see *About Responsibilities*.

About Responsibilities

Responsibilities provide access to specific Views to particular users. Different types of users need access to different views for varying functional reasons. For example, a sales representative should not be working on service requests and a field service engineer should not be creating marketing campaigns for visibility reasons or a partner user should not be able to see sales opportunities that partners from a different company are working for competitive reasons.

In addition to security and functional reasons, Responsibilities ensure that users are able to focus on the information most relevant to them. Responsibilities allow you to present only those views that a user actually needs to do their job by providing a many-to-many relationship between Views and Users.

The easiest way to conceptualize and define Responsibilities is to consider generic job classes. For example, sales representative, sales manager, service technician, or even CEO. These are classes of job that require access to different types of data. For example, sales users need access to sales-related data, such as sales opportunities, while service users need access to service-related data, such as service requests and require access to different levels of data and sales representatives need access to the data of customers that they directly work with, while service users need access to service data for all customers.

About Seed Data Responsibilities

Seed data responsibilities provide access to out-of-the-box views for various user types that are found in a given organization. There are over 400 Responsibilities defined and there are over 100,000 mappings from these responsibilities to various Views. While this sounds like a lot, consider all the different types of users within a large organization and the records to which they might need access:

- **Sales Representatives.** Sales representatives need access to Accounts and Opportunities that they are working on.
- **Sales Managers.** Sales managers need access to Accounts and opportunities that they or their reports are working on.
- **Marketing Managers.** Marketing managers need access to all customers in order to conduct marketing campaigns.
- **Field Service Engineers.** Field service engineers need access to all customers associated with service requests that they are working on.
- **Call Center Support Engineers.** Call center support engineers need access to all customers since any customer can call and service requests they are working on.
- **Fulfillment Center Workers.** Fulfillment center workers need access only to requests for literature fulfillment and the contact information for the recipients.
- **CEOs.** CEOs need access to all customer data. However, the CEOs may have restrictions on the types of changes that they can make and should not be able to modify system administration data. For example, changing Siebel CRM application behavior.
- **Application Administrators.** Application administrators need access to everything so they can address any issues that may occur.

For these reasons, Oracle engineers have seeded responsibilities to reflect the most common roles seen within an enterprise. This makes it easy to quickly associate a given new user with an existing Responsibility.

Adding New Responsibilities

There are many seeded Responsibilities, but nearly all Siebel CRM customers add their own Views and need a way to expose those Views to their users. The recommended approach is to use a combination of seeded Responsibilities and custom Responsibilities.

- Giving a user a seeded Responsibility will ensure that a user receives new views relevant to their job function as and when Oracle releases new functionality within the product. For example, assume that Oracle Engineering adds a new set of views to the existing sales functionality in Siebel CRM. Oracle Engineering will not just create the views, but will also add them to appropriate existing Responsibilities. Therefore, a customer user with that Responsibility will receive access to the new functionality whenever that new release is deployed without any administrative changes being required.
- Custom Responsibilities will contain all views developed by the customer and any out-of-the-box views that do not come as part of their out-of-the-box Responsibilities.

To prepare for associating users with the correct Responsibilities, perform the following steps:

1. Determine the distinct job classes that will exist in your organization. For example, these might include Field Sales Representative, Sales Manager, Marketing Manager, Call Center Agent, or Call Center Manager.
2. Examine the seeded Responsibilities and determine which are appropriate for each job class.

   In many cases, a manager job class may have the same Responsibility as the subordinates and an additional Responsibility for the additional views required for their management responsibilities.
3. Plan to create an additional Responsibility for each user class. This will contain all custom views for that user class. The following table describes the out-of-the-box responsibilities and custom responsibilities for several user classes.

User Class	Description	Out-of-the-Box Responsibilities	Custom Responsibilities
Sales Representative	Individual contributor, works with a specific set of customers to close sales deals	Field Sales Representative	XYZ Field Sales Representative
Sales Manager	Manages a team of Sales Representatives.	Field Sales Representative Sales Manager	XYZ Field Sales Representative XYZ Sales Manager
Marketing Manager	Plans and manages marketing campaigns.	Marketing Manager	XYZ Marketing Manager
Call Center Agent	Answers phone calls from customers requesting service.	Call Center Agent	XYZ Call Center Agent

User Class	Description	Out-of-the-Box Responsibilities	Custom Responsibilities
Call Center Manager	Manages the call center and its employees	Call Center Agent Call Center Manager	XYZ Call Center Agent XYZ Call Center Manager
Siebel Administrator	Manages all aspects of the Siebel CRM application.	Siebel Administrator	XYZ Siebel Administrator

These examples assume your company is XYZ. Use any simple prefix that will allow you to easily determine which are your custom Responsibilities and make them easy to find when querying.

4. Having documented the Responsibilities you plan to use, navigate to **Site Map > Administration - Application > Responsibilities** to add the custom Responsibilities. Since you have not created any custom views, yet, it is possible that your custom Responsibilities will have no related Views, which is expected. You will associate the Views to these custom responsibilities later as they are developed.

About Organizations

In Siebel CRM, Organizations allow data to be segregated across divisions or other organizational units within the overall company. Many companies do not leverage Organizational visibility in Siebel CRM, but it is a useful tool to segment data across sets of users. The definition of an Organization is at the discretion of the company implementing Siebel CRM, but some examples of reasons why Organizations might be appropriate include the following:

- **Disparate geographies:** Users in North America should not see Opportunity data from Asia Pacific. This could be for external legal or regulatory reasons or internal governance reasons.
- **Partner visibility to data:** Consider a company who sells its products through external partner companies who use Siebel CRM. For competitive reasons, partner companies should not be able to see sales leads belonging to other partners. Segmenting data organizationally by partner ensures that this could never occur.
- **Diverse product areas:** Consider a conglomerate that sells jet engines to airplane manufacturers and home appliances to end-consumers. Every type of data such as products, sales representatives, sales channels, potential customers, and pricing are all completely unrelated to each other and should not overlap.

Every company should consider its own needs and determine whether Organizational visibility makes sense. If so, it is best to create Organizations in advance as it is easier to ensure data is correctly segmented at the time it is created. Creating Organizations in advance is recommended even in Development and Test environments as this ensures that developers are always aware of the Organizational visibility requirements and are inherently developing and testing for it.

To create an Organization, navigate to **Site Map >Administration - Group > Organizations** and create new records based on your business needs. For more information about setting up Organizations, see *Siebel Security Guide*

About Positions

Positions represent job slots within an organization, typically a sales organization. Consider a sales organization that is defined by region. For example, Northeast Florida sales representative, South Florida sales representative, or Florida Panhandle sales representative. At different points in time, a different human may be handling that region. John Smith may be responsible for Northeast Florida today, but he might be moved to South Florida, promoted to sales manager, or even leave the company.

The customers and potential customers in his region do not change just because John moves to a different position or company. Someone else, perhaps Dave Jones, will take over those customers and opportunities and continue to manage that territory.

> **Note:** Territory does not necessarily refer to a geographic region. It could also be broken down by other factors, such as deal size, or the types of products involved in an opportunity. In this sense, the term Territory is used generically to represent some logical division of target customers and opportunities as appropriate to the company.

For that reason, Siebel CRM does not associate sales data, such as customer accounts, contacts, sales leads, or opportunities, with specific users, but rather with the position that conceptually owns this data.

At a given point in time, a given user will hold a position, in this example, Northeast Florida Sales Representative and be able to see all the relevant data for that position. When John Smith moves on, he is disassociated with that position and the new user, Dave Jones, is associated to that position. This immediately moves all the data visibility from John to Dave.

The following are some guidelines for creating positions:

- For sales-centric organizations, there should usually be a one-to-one mapping between positions and users. You should create a position for each user before creating the actual users. There will be occasions where a user might have two positions. For example, John Smith is on vacation or the organization is looking for his replacement, so Dave Jones temporarily has his own position and John's, but as ideally the organization would like two sales representatives for this, two separate positions should be maintained in this type of example.
- For service-centric organizations, positions play a much smaller role, because a service user is typically not bounded by geography or other constraints. For example, if a user works in a call center, answering phone calls from customers, that user is not typically associated to a particular region or set of customers. Rather, they must assist any customer who calls in. As such, in a strictly service organization, it is common for service users to share a position.

It is possible to have a combination model within a single company–individual positions for sales users based on the way in which territories are assigned while service users share a position.

> **Note:** In some sense, positions and organizations have similar purposes. However, it is just a difference in granularity. For example, if an organization is used to segregate data at the continent level (North America), a position is used within that organization to represent some subset, such as Northest Florida. As such, organization visibility typically includes data visible to a group of people, while position visibility includes data visible to an individual and his or her management.

Before continuing forward, create the required position records using the guidelines for determining whether individual positions are required for each user or whether positions can be shared. Positions are created by navigating to the following: **Site Map > Administration - Group > Positions**.

About Users and Employees

Users represent individual humans (or, in the case of integration, virtual humans) who use the Siebel CRM system. A User is any valid user (person or automaton) that can use the system, and has a minimum requirement of a first and last name, Login Id, and a Responsibility. An Employee is a special class of user who additionally must have a Position and may also have other optional attributes, such as skills and certifications.

The distinction of Employees (having a Position) is the key difference–as already noted, sales-related data (sales accounts, opportunities, and contacts) are associated to Positions, and therefore only Employees (through their Positions) can own such data. Non-employee users, such as customers who are registered to use the system to log service requests or place sales orders, cannot own that type of data.

> **Note:** A Siebel CRM Employee is not necessarily an employee of the organization. There are examples of non-employees who may be registered as Employees, such as temporary contractors or partners who need to be able to own sales-related data.

When creating a User record, consider the following:

- **Responsibilities (groups of views).** Users require one or more Responsibilities. This determines which Views that user can even reach. It is important to understand that Responsibilities and Views dictate where the user can go. Responsibilities and Views do not fundamentally determine which data a user can see. For example, assume that a user, through his Position, can see Contact data for a particular customer. Unless the user also has access to a Contact view through a Responsibility, that user has no way to see that data, even though they are entitled to see it. For more information about responsibilities, see *About Responsibilities*.
- **Organizations.** An Employee User may be associated with one or more Organizations. This allows the Employee to readily access in their own Organization, for example, account records in their geographic reason, while preventing them access to data outside that region. For more information about organizations, *About Organizations*.
- **Position.** An Employee User may have one or more Positions. As with Organizations, Positions control visibility to individual records within the application, such as specific customer accounts, but at a more granular level. It should be noted that while an Employee may indeed have multiple Positions, a user can only actively use one at a time. For more information about positions, see *About Positions*.

About Seed Data Users

While Oracle cannot predict what Users will use a particular Siebel CRM instance, there are several Users, some of which are also Employees, that are created as Seed Data to serve specific functions. This topic will explain those Users, as well guidelines for adding new Users.

The following Users are required for the Siebel CRM system to function:

- **SADMIN (Employee User).** After installing Siebel CRM, it is necessary to be able to log in as some user in order to do anything. This is true of any system. For example, a new Microsoft Windows implementation always has an Administrator user. While a Unix system will have an admin user. The SADMIN user is that lynch pin user for

Siebel CRM. It is loaded as part of seed data and by default has access to every screen and view in the Siebel CRM application.

It should be noted that there is a common misconception that SADMIN is somehow magical, in the sense that it can circumvent various security mechanisms. For example, there is an assumption that if a user creates a new customer account record, SADMIN will automatically be able to access that record. That is not accurate. The SADMIN user has access to the Account Administration View, which in turn provides access to all customer Accounts; however, were the SADMIN user to lose access to that Administration View, it would lose access to customer accounts to which it is not directly associated.

However, this is precisely why Seed Data associated with the SADMIN user should not be modified. For example, removing the SADMIN user's access to particular parts of the application or certain records could result in a situation where the application is no longer configurable or data is essentially unreachable, because no user has access.

Therefore, rather than adjust SADMIN's default access, it is preferable to restrict who can log in as this user by keeping the password secret.

- **GUESTCST (Customer User), GUESTERM (Employee User), and GUESTCP (Partner User).** These are proxy users. By default, they have access to very little, but what they can access is very important, as they determine what users can see before they login. Consider, for example, an Internet-facing customer portal for a bank. That bank may use Siebel CRM for customer-specific activities, such as checking account balances or initiating payments. However, it will also offer basic services that should be available without the need for a customer login. For example, a user should be able to visit the web site and look up the location of the nearest branch of the bank and should not have to log into their account to do so.

  The proxy users in this example, GUESTCST, provide this capability. Each has access to a set of screens and views within the application that should be readily available without logging in.

  There three different users allow some flexibility in terms of what type of user can see which views without logging in. Any can be specified as the Anonymous User when creating a Siebel Application Interface Profile as explained in the Installing the Siebel CRM Environment chapter. In the out-of-the-box seed data, these are mapped to out-of-the-box views as follows:
- **GUESTCST (Anonymous Customer User).** Includes access to customer-facing views, such as branch locator or product configuration.
- **GUESTERM (Anonymous Employee User).** Employee facing views, such as support knowledge bases for users to browse before creating a HelpDesk Service Request.
- **GUESTCP (Anonymous Partner User).** Partner facing views, which could include any or all of the previous views, depending on the particular use case.
- **PROXYE.** This is used during the User Self-Registration process. For example, a customer who wants to register for a site hosted by Siebel CRM.
- **UNIVERSALQUEUE**. This user is obsolete. It can be ignored as it has no ability to access the application because this user has no way to log into Siebel CRM.

About Creating New Users and Employees

While it is possible to create non-employee users within the Siebel CRM application, most non-employee users are created by the actual user through self-registration. For example, a customer who wants to place an order through the sales application registers before being able to complete the order, or an existing customer creates an account the first time they log a service request. The process of self-registration and how to customize the user experience is documented in the *Siebel Security Guide* . This topic focuses only on creating new Employee records.

There are two high-level steps for creating an employee user:

- Setting the employee's Siebel CRM attributes.
- Setting the employee's authentication requirement.

On the Siebel CRM side, creating an employee user involves the following steps:

- To ensure the most flexible visibility scheme, Oracle recommends that each employee has a unique position. This is not a strict requirement for employees. It is required for sales users. It is also possible for service users to share positions. Adding a position provides for finer control over data visibility and there is no downside to creating a unique position for each user. The first step in creating a new Employee record is to first create a Position record.
- After creating Position records for each new employee, navigate to **Site Map > Application > User > Employee**.
- Create a record with the user's first name, last name, and a login name (this may be arbitrary or may be dictated by the authentication mechanism). While not required, it is strongly recommended that each employee have an email address, as this can be leveraged for integration with corporate mail systems such as Microsoft Exchange, as well as sending email from within Siebel CRM.
- Associate the Employee's Position and specify one or more Responsibilities.

Having completed these steps, Siebel CRM now considers the Employee a full-fledged user. This user can be associated to Siebel CRM objects, such as Accounts, Opportunities, Contacts, or Service Requests.

However, the user will not actually be able to log into Siebel CRM. The reason for this is that Siebel CRM does not provide a native method for authentication, but rather relies on external authentication mechanisms. These are described in detail in the *Siebel Security Guide* , but are summarized here:

- **Database Authentication.** When database authentication is configured, Siebel CRM passes the username and password provided by the user directly to the underlying database for authentication. If you use this mechanism, each user must be created in the database by the Database Administrator and have access to the Siebel CRM database objects, such as tables and indices.

  > **Note:** This is not the recommended model because a user can connect directly to the database through a database-specific utility, such as SQL*Plus, and would be able to interact freely with the data in the database. This creates a potential security problem.

- **LDAP Authentication.** With LDAP authentication, Siebel CRM uses an external LDAP authentication authority to validate a user's login credentials. The advantage of this approach over Database Authentication is the following:
  - A user can use their same credentials across many systems that share a common LDAP authority.
  - Because users do not have direct database credentials, the security issue described with Database Authentication does not apply.
- **SSO Authentication.** SSO authentication is very similar to LDAP authentication in that an external authentication authority is used to validate the user's login credentials. The difference is that under SSO authentication, once a user has logged into any SSO-enabled application within the enterprise, the user is not required to login again as long as the browser is open. For example, consider the case where Siebel CRM shares an SSO authentication authority with two other systems, X and Y. When a user navigates to system X, he or she will be required to enter credentials. If that same user opens a new browser tab and navigates to Siebel CRM (or system Y), the user will not have to provide credentials a second time.

Regardless of the authentication mechanism in use, to allow the Employee User to have access to the Siebel CRM system, it will be necessary to provision the user in that authentication authority. In the case of Database Authentication, this would mean creating the user in the database. In the case of the other two authentication options, it would mean provisioning the user in that authentication authority and providing a mapping from that user to the Siebel CRM login. For example, mapping SSO user john.smith@company.com to Siebel CRM user JSMITH. For specifics relevant to your implementation, see the *Siebel Security Guide* .

About Creating Developer Users

A *Developer User* is a special class of user that is empowered to make configuration changes to the Siebel CRM application. Historically, a developer user was a traditional technical resource who performed coding and configuration changes. With the release of Siebel Innovation Pack 2017 and Siebel Web Tools, the ability to configure the Siebel CRM application can be extended to a wider set of users such as developers and business analysts. Basic changes such as adding a field or re-organizing the user interface in Siebel CRM can be completed by a much wider community of users.

The ability to modify the Siebel CRM application should not be granted lightly. When you modify the Siebel CRM application, keep in mind the governance process and the use of Siebel Approval Manager to ensure that changes are properly vetted and approved. Leaving aside the governance process, the basic capability for a developer or business analyst requires the following:

1. Each user who will be allowed to make configuration changes must be granted the Composer Administrator responsibility.
2. Each user who will be allowed to deliver configuration changes to the main branch must be granted the Workspace Administrator responsibility.
3. Any Developers or Business Analysts who will be making changes needs access to all views in the Siebel CRM application. For example, to create new View and Responsibility records or testing their changes to all objects. In a development environment, developers or business analysts must be granted the Siebel Administrator responsibility.

After granting any new responsibility to a user, the user must log out of the Siebel CRM application and close their browser in order to see the changes made to Siebel CRM.

> **Note:** It is not the best practice to routinely log in as the SADMIN user as this effectively eliminates any way to audit who makes changes to the Siebel CRM application. Granting the Siebel Administrator responsibility to users gives those users equivalent capabilities within the Siebel CRM application while allowing an accurate audit trail of changes.

About Other Seed Data

The critical seed data items that you will need to allow users to use the system while providing access to those functional areas and data they need while preventing them from others are addressed by the Seed Data objects. There are a number of other common Seed Data objects that will also need customization in order to meet the business requirements for your particular organization. The most common of these include:

- *About List of Values*
- *About Time Zones*
- *About Locale*

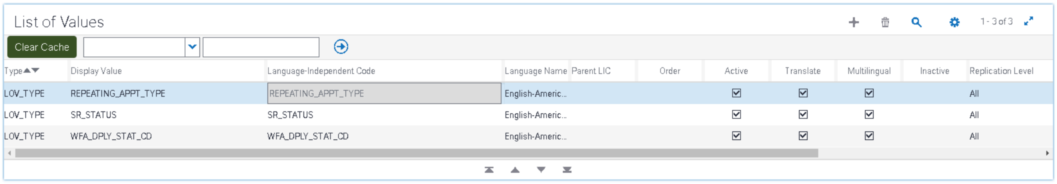
- *About Currency*
- *About Sales Methodology and Sales Stages*

About List of Values

The List of Values (LOV) entity is one of the most important in the Siebel CRM application. It provides the values in almost every drop-down or picklist within the Siebel CRM application. For example, setting the status of a Service Request to New, In Progress, or Waiting on Customer, or the recurrence pattern for a Calendar appointment to Weekly, Daily, or Monthly. LOV's also provide values to even more internal technical items, such as the deployment state of a Workflow Process, which are all made available in the List of Values Seed Data.

A LOV is simply a huge list of all the drop-down values throughout the Siebel CRM application that are broken out by a type, that is stored in the LOV known as LOV_TYPE. The LOV_TYPE filters each individual picklist. Using the previous examples of Service Request Status, Calendar Recurrence, or Workflow Process Status, you will see that there are three different LOV_TYPEs as follows and shown in the following image:

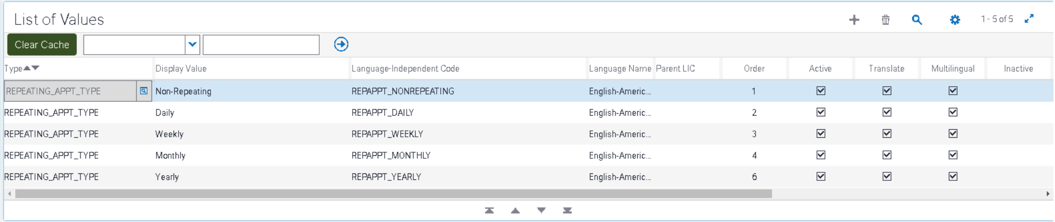
- **REPEATING_APPT_TYPE:** The Calendar Recurrence LOV_TYPE.
- **SR_STATUS:** The Service Request LOV_TYPE.
- **WFA_DPLY_STAT_CD:** The Workflow Process Status LOV_TYPE



List of Values

Type	Display Value	Language-Independent Code	Language Name	Parent LIC	Order	Active	Translate	Multilingual	Inactive	Replication Level
LOV_TYPE	REPEATING_APPT_TYPE	REPEATING_APPT_TYPE	English-Americ...			☑	☑	☑		All
LOV_TYPE	SR_STATUS	SR_STATUS	English-Americ...			☑	☑	☑		All
LOV_TYPE	WFA_DPLY_STAT_CD	WFA_DPLY_STAT_CD	English-Americ...			☑	☑	☑		All

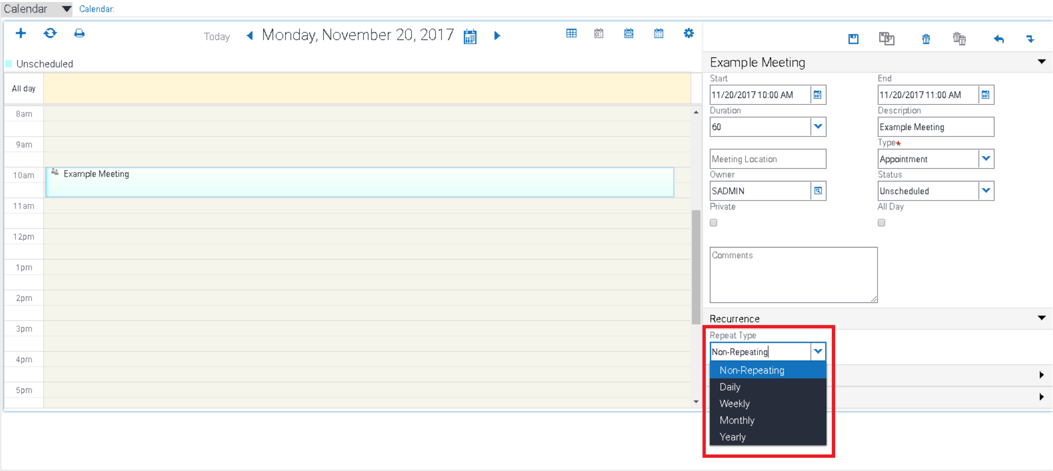
Each LOV_TYPE has specific records/values relevant to the picklists where they are used. For example REPEATING_APPT_TYPE has the following records/values, as shown in the following image: REPAPPT_NONREPEATING, REAPPT_DAILY, REPAPPT_WEEKLY, REPAPPT_MONTHLY, and REPAPPT_YEARLY.



List of Values

Type	Display Value	Language-Independent Code	Language Name	Parent LIC	Order	Active	Translate	Multilingual	Inactive
REPEATING_APPT_TYPE	Non-Repeating	REPAPPT_NONREPEATING	English-Americ...		1	☑	☑	☑	
REPEATING_APPT_TYPE	Daily	REPAPPT_DAILY	English-Americ...		2	☑	☑	☑	
REPEATING_APPT_TYPE	Weekly	REPAPPT_WEEKLY	English-Americ...		3	☑	☑	☑	
REPEATING_APPT_TYPE	Monthly	REPAPPT_MONTHLY	English-Americ...		4	☑	☑	☑	
REPEATING_APPT_TYPE	Yearly	REPAPPT_YEARLY	English-Americ...		6	☑	☑	☑	

These values appear in the application when selecting the repeating pattern for a meeting. The following REPEATING_APPT_TYPE values appear in the Repeat Type picklist, as shown in the following image: Non-Repeating, Daily, Weekly, Monthly, Yearly.



Example Meeting

Start: 11/20/2017 10:00 AM
End: 11/20/2017 11:00 AM
Duration: 60
Description: Example Meeting
Meeting Location
Type*: Appointment
Owner: SADMIN
Status: Unscheduled
Private □
All Day □
Comments

Repeat Type: Non-Repeating

The LOV_TYPE value in this example, REPEATING_APPT_TYPE, determines which items show up in which picklist based on settings in the Siebel Runtime Repository.

There are two classes of LOVs in the Siebel CRM system. There are LOVs that can be safely modified and those that cannot be modified. There is no hard rule for this, but before modifying an LOV, administrators should consider if there is specialized functionality that might be based on the LOV. Consider the three LOVs that are used as examples here: REPEATING_APPT_TYPE, SR_STATUS, and WFA_DPLY_STAT_CD.

- **REPEATING_APPT_TYPE:** This LOV type indicates the types of repeating patterns that the Siebel Calendar can support. Logically, adding a new one, for example, every hour, is not going to work because the underlying calendar code will have no idea what to do with an hourly appointment. Oracle does not recommend customizing this LOV type.
- **SR_STATUS:** This LOV type indicates the current status of a Service Request. This is an example of an LOV that can be modified, but there are some considerations. Specifically, based on the underlying object definition, it is not possible to make changes to a Service Request once it is closed. Therefore, while it is possible to add to the list of status values available, there could be negative behavior were the administrator to remove or rename the Closed status.
- **WFA_DPLY_STAT_CD:** This is an example of an LOV that must never be modified. It is controlling the behavior of a repository object. The status of a given Workflow Process can certainly change, the administrator can mark a Workflow Process as Inactive or Active. However, making changes to the possible values would have unknown and undesirable repercussions.

These examples all imply that it is not allowed or at least not recommended to change LOVs but that is not the case. For example, there is no reason not to modify Account or Contact Status values, the Service Request Areas and Sub-areas since these are completely acceptable changes.

Some guidelines on changing LOVs:

- As already noted, do not change LOVs for objects that control system behavior, such as Workflow Process Status.
- Be careful modifying highly specialized objects or those that illustrate unique behavior in the application. For example, if you note that closing a Service Request makes it read-only, then be careful modifying that LOV type.
- As with all seed data, do not delete LOVs. If something is not desired in your implementation, inactivate the LOV. On a side note, doing so is not effective in the long run since Siebel CRM upgrades restore missing LOVs. Removing a LOV lasts until the next upgrade, while inactivating it will last forever and the upgrade process will not reactivate inactive LOVs.

- Similar to the last point, do not rename LOVs. If a name change is desired, for example changing In progress to Working. In that example, it would be best to inactivate the undesired LOV and create a new active one. Failure to do so will result in the renamed LOV coming back at the next upgrade and may throw errors during the upgrade process.

> **Note:** There are multilingual options for LOVs. These are discussed previously in the chapter on installing the environment as well as the *Siebel Global Deployment Guide* .

About Time Zones

Time zones are loaded as seed data into the Siebel CRM database and are used for the following purposes:

- There are various customer dashboards used within the service areas of the application that allow a user working with a customer, such as a live phone conversation, to see information about that customer, including their current time.
- The calculation of Service Level Agreements. For example, if a customer is guaranteed a response within a certain number of hours, it is important to understand the time zone of the customer to accurately calculate how long the issue has been open or waiting a response.
- Integration with other systems. For example, consider integration with Microsoft Exchange for calendar items. Without a common understanding of time zones, meetings may appear to shift as they are synchronized between the two systems.

In general, time zones are updated in every major Siebel CRM release, including every Innovation Pack. However, time zones change due to legislative reasons on a regular basis. For example, the United States changed the start and end dates for Daylight Savings Time (DST) in 2005. Other countries make more frequent and irregular changes, such as Brazil, which moves the DST transition date around based on Mardi Gras.

If any of these types of concerns are relevant to your implementation, ensure that Time Zones are updated regularly.

About Locale

Locale is a combination of geography, language, and culture. To understand Locale, consider the difference between the Unites States and the United Kingdom. Both share a common language, but they use completely different units of measure. The United States uses the English language, but use miles, degrees Fahrenheit, and gallons, while the United Kingdom uses kilometers, degrees Celsius, and liters.

Other examples include parts of Belgium and France that use French as a national language, but they have different dialects. For example, ninety is quatre-vingts dix in French, but nonante in Belgium.

Locale also covers punctuation. North Americans, for example, use a comma to separate the digits in a large number (1,000,000 being one million), while central European countries use a period (1.000.000 being the same million in central Europe) and Switzerland uses an apostrophe for the same million: 1'000'000.

Oracle provides seed data that reflects the common locales, but it is important that each locale in which you will have users is double-checked for accuracy.

About Currency

Currencies do not often change. The Euro is the last major change in currency. However, currency exchange rates constantly fluctuate. If your Siebel CRM application will be doing calculations such as forecasts, orders, or billing, in more than one currency, it is important to have a system in place for constantly updating the current exchange rate between various currencies.

> **Note:** Out-of-the-box Siebel CRM objects include the capability of specifying an exchange date for a transaction, thereby allowing an exchange transaction to be calculated at the date that a transaction occurs. However, without a constant update to the currency tables to reflect current exchange rates, this capability will provide inaccurate information. For this reason, customers who need to do currency conversion should develop a plan for maintaining currency exchange rates on a given date.

About Sales Methodology and Sales Stages

For sales organizations, a common option is the selection of a sales stage. For example, a particular Opportunity might be Unqualified, Qualified, Lead, or Short List. In Siebel CRM, this concept is handled at two levels. The Sales Methodology is a parent attribute that defines the process followed for an Opportunity. The sales stages are the children relevant to the type of Sales Methodology.

There are some sales processes that require many steps. For example, a sale to a large corporation with a procurement organization that requires multiple sign-offs. or a fewer number of steps where a less complex process is needed to work through the sale. These are seen on Sales Opportunities. The following table contains examples of sales methodologies:

Sales Methodology	Sales Stages
Default Sales Methodology	• 01 - Prospecting • 02 - Potential Lead • 03 - Qualification • 04 - Opportunity • 05 - Building Vision • 06 - Short List • 07 - Selected • 08 - Negotiation • 09 - Closed/Won • 09 - Closed/Lost
Mortgage Application	• 01 - Qualification • 02 - Requirements

Sales Methodology	Sales Stages
Strategic Selling	• 01 - Universe • 02 - Above • 03 - In the Funnel • 04 - Best Few • 05 - Won 00 - Lost

About System Preferences

The System Preferences control the overall Siebel CRM application behavior. In general, changing System Preferences effects every user unless there exists a specific user-level override, which is unusual. Individual System Preferences are documented throughout the Siebel Bookshelf by functional area. It should be noted that most System Preferences take effect at the next Siebel Server restart.

5 Customization in Siebel CRM

Customization in Siebel CRM

Customization in Siebel CRM requires an understanding of Siebel Repositories and Workspaces. This chapter includes the following topics:

- *About Customization in Siebel CRM*
- *Siebel Repositories*
- *Siebel CRM Workspaces*

About Customization in Siebel CRM

Siebel CRM is extremely customizable, allowing you to modify and create new objects, properties, validation rules, workflow, guided help (iHelp), integration with other applications, and so on.

This customization requires an understanding of the following:

- **Siebel Repository.** Contains metadata that defines how the application behaves. For example, it defines all of the tables in which data is stored, the business logic layer, the objects that allow integration with other systems, the user interface, and so on. The Siebel CRM application provides an extensive set of repository metadata, providing you with functionality supporting all aspects of customer relationship management across many industries and 22 out-of-the-box languages with the ability to add more. Customers can modify and augment the repository to support their business needs. For more information, see *Siebel Repositories*.
- **Workspaces.** Provide a version-controlled mechanism for business analysts and developers to make changes to the Siebel Repository. It allows for many users to make changes simultaneously and in parallel, for example, when they are working on unrelated features, different timelines, or for other reasons. Workspaces provide standard version control features, such as checkpointing, delivery, reversions, and rebasing. Workspaces replace the older process of checking in and out that was previously used in Siebel Business Applications. In Innovation Pack 2017, it is mandatory to use Workspaces. This provides more flexibility, parallel repository development, and the ability to integrate with Siebel Approval Manager (or other tools) for change control. For more information, see *Siebel CRM Workspaces*.

Siebel Repositories

Technically speaking, there are actually two Repositories involved in the implementation of Siebel CRM.

- **Siebel Design Repository (DR).** This repository contains the editable definitions of the metadata. Siebel Tools and Siebel Web Tools applications use this repository to make changes to the repository.
- **Siebel Runtime Repository (RR).** This contains a compiled, denormalized version of the Siebel Repository. Siebel Application Object Managers use this at runtime to provide application functionality to users and integration points. This in-database version of the repository replaces the old Siebel Repository File (SRF). Changes to the runtime repository affect application behavior in real-time, unlike the old SRF paradigm, which

required compilation and restarting Application Object Managers and other components for changes to take effect.

For information about managing repositories, see *Using Siebel Tools* .

About the Siebel Design Repository

The Siebel Design Repository is the traditional repository that has always been part of Siebel CRM. It contains editable metadata that defines application behavior. It is used make modifications to the delivered Siebel CRM applications.

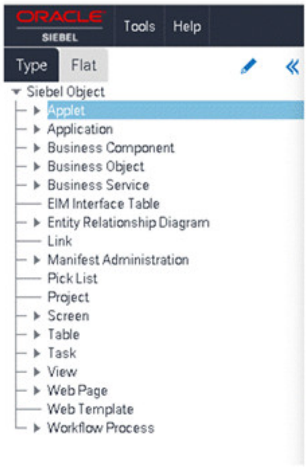
To configure these modifications, you make changes to the Siebel Design Repository by using either one, or both, of the following tools:

- **Siebel Web Tools.** This allows configuration of repository objects within a Web browser.
- **Siebel Tools.** This allows any functionality that has not yet been moved to Siebel Web Tools. Developers usually perform this level of configuration. Eventually, all functionality remaining in Siebel Tools will be migrated to and only available in Siebel Web Tools.

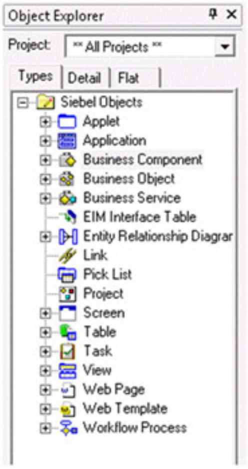
The Siebel Design Repository consists of a hierarchical set of objects. For more information, see *Siebel Object Types Reference* , but let's consider some simple examples:

- **Table Objects.** Describe the physical tables in which data is stored. Table objects and their child objects are often referred to as the data layer.
  - **Column Objects.** Describe the individual data attributes within those tables.
  - **Index Objects.** Describe database indices used for optimization, uniqueness, or performance.
- **Business Component Objects.** Provide an abstraction layer on top of the data layer. These are often referred to as part of the business layer that helps define relationships between data as well as business rules affecting that data. This allows for far more complexity than you define at the database layer.
  - **Field Objects.** Describe data elements and basic validation rules, for example, which fields are required, numeric, read-only, and so on.
  - **Join and Multivalue Link Objects.** Define relationships between objects, for example, a Contact object can have many Address objects.
- **Applet Objects.** Provide a visual representation of a business component so that a user can interact with it, and can then form part of the user interface layer.
  - **Control Objects.** Define the possible items that can appear on the applet, for example, buttons, labels, fields, menus, and so on.
  - **Web Templates.** Provide possible physical layouts for the applet, that is, specifying controls and their corresponding locations.

There are 50 top-level repository object types, many of which have child objects as described in the previous examples. By default, only the most commonly used object types (for example, Applet, Application, Business Component, Business Object, Business Service, Task, View, Workflow Process, and so on) are displayed in Siebel Tools and Siebel Web Tools, as illustrated in the following image.



Siebel Webtools



Siebel Tools

To expose or remove object types from being available, do one of the following:

- **Siebel Web Tools.** Click the pencil icon.
- **Siebel Tools.** Click Options, then Object Explorer in the View menu.

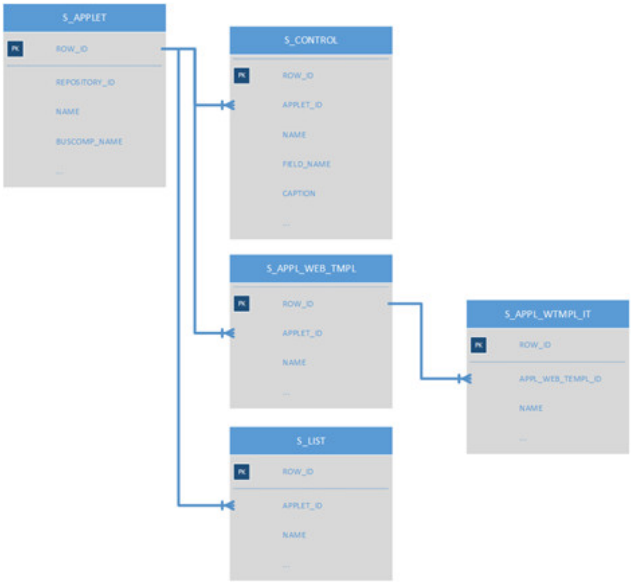
About the Siebel Runtime Repository

Whether operating as the user interface, or operating in the background as an interface with other systems, how the Siebel CRM application behaves depends on the contents of the Siebel Repository.

In theory, the Siebel CRM application could read its configuration directly from the Design Repository tables, and then to function according to this specific configuration. This approach, however, is inefficient. These tables store hierarchical data with great-grandparent objects, grandparent objects, parent objects, and so on, and pulling information from the supporting tables at runtime would inhibit performance at runtime.

Consider the following diagram, which illustrates a basic Applet object, its corresponding entity relationships, and its layout, as physically defined in the database.

> **Note:** This diagram only identifies some of the Applet objects attributes, and, for example, does not illustrate foreign keys to other objects.



The purpose of this diagram is to illustrate only that the top level (or parent) object has many child objects, and that there are relationships from each child object to the parent, as well as relationships from the grandchild to the child objects, and so on.

When you create applets, make changes to existing applets, and so on, the underlying tables in the Design Repository are updated to reflect those changes.

In its entirely, the Applet object has a top level object based on S_APPLET, and 13 child objects (Controls, Lists, Charts, Web Templates). Some of child objects have grandchild objects (for example, Web Template Items), which add a further thirteen tables to the applet. In addition, there are another six great-grandchild objects, these are mostly locale-specific captions. To get a full definition of an applet from the Design Repository at runtime, would require the Siebel CRM application to query 33 tables.

To improve runtime performance, the Siebel Design Repository data is compiled into the Siebel Runtime Repository. This process summarizes each object type into an easily queried set of tables that has a one-to-one mapping to the top- level repository objects. For example, an Account List Applet has 822 distinct records across 33 tables in the seeded repository. However, in the Siebel Runtime Repository, these 822 records are condensed to a single denormalized, compiled record that summarizes all 822 records and is placed in a runtime repository table, for example S_RR_APPLET.

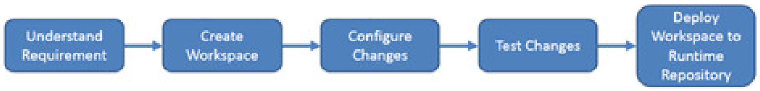
Siebel CRM Workspaces

Understanding Workspaces is critical to modifying the Siebel Design and Siebel Runtime Repositories, and to accommodating specific customer business requirements effectively.

A Workspace is a sandbox environment in which you can update Siebel CRM application configuration, and in which you can validate those configuration changes before deployment. You can also use Workspaces for parallel development, geographically distributed teams, and, in conjunction with Siebel Approval Manager, providing a governance process to changes to the application.

We will illustrate the basic use of Workspaces through a simple example: you want to display contact home phone numbers instead of work phone numbers on the Contact List Applet.

The steps in the configuration process, as illustrated in the following image, are as follows: Understand Requirements, Create Workspace, Configure Changes, Test Changes, Deploy Workspace to Runtime Repository.



Before diving into configuration changes, it is important to have an understanding of the scope of the requirement and have a clear plan for addressing it in the most appropriate way. By way of example, there are multiple ways that this requirement could be met, including the following possibilities:

- Modify the label on the existing column in the list applet to be: Home Phone #. Analysis: While this approach would work, the underlying field and physical database storage would still be using the Work Phone # field. There may be other areas of the application or other types of users who want to continue to use this field for the Work Phone #, so relabeling it here would cause confusion and possible data loss.
- Create a new physical column at the database layer, map a BusComp Field to that column, and add it to the list. Analysis: This would also work, but by examining the existing Table and BusComp definitions, we can observe that a column already exists for this purpose. Creating a new column would mean that home phone numbers could be stored in two different columns–the original and our new column. This is clearly not desirable.
- Make use of the existing Home Phone # field on the Contact BusComp. Analysis: This is the best approach–it will ensure that the types of problems inherent in the first two approaches are not encountered. It will also be the fastest and easiest, as there is no requirement to make changes at the data or business object layers–all changes are done at the user interface layer.

Having performed our analysis, we can confine the scope of the change needed to meet the requirement to the following two changes:

- Expose the existing Home Phone # Field on the Contact List View so that users may see and edit it.
- Remove the existing Work Phone # Field

For more information about Workspaces, the Workspace Dashboard, and how to use and administer Workspaces, see *Using Siebel Tools* .

6 Adding Additional Languages

Adding Additional Languages

Oracle recommends adding languages before performing any configuration of the Siebel CRM application. This chapter provides overview information about how Siebel CRM handles working with multiple languages in the same Siebel CRM instance and adding additional languages to Siebel CRM. It includes the following topics:

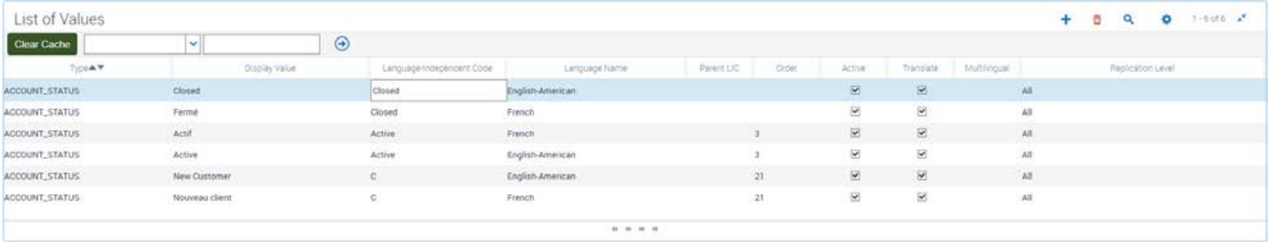
- *About Language and Seed Data*
- *About Languages and Repository Data*
- *About the Process of Adding Additional Languages*

About Language and Seed Data

When you use Siebel CRM in a particular language, you see the values for that language. For example, when selecting the status for a customer account record, an English user sees values such as New Customer or Active, while a French user sees Nouveau client or Actif. For this behavior to work, LOVs must be converted to MultiLingual List of Values (MLOV) through the MultiLingual List of Values conversion process.

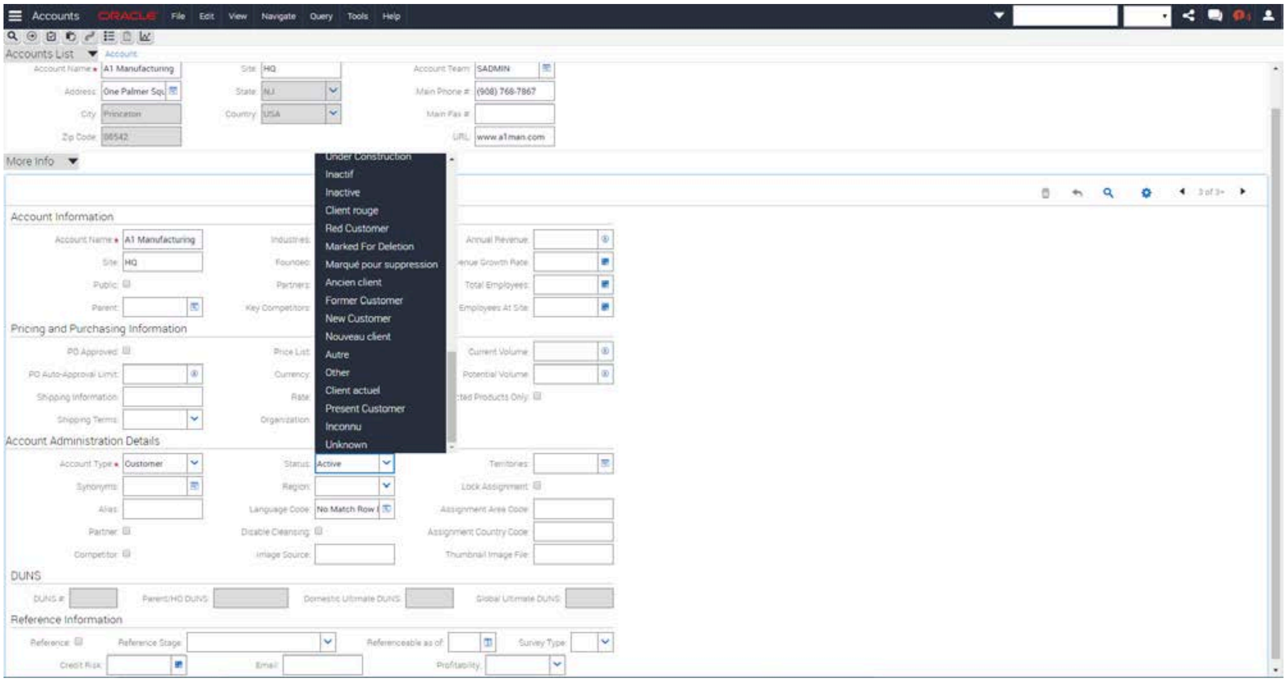
To understand how the MLOV conversion process works, consider how Siebel CRM stores data when MLOV conversion has not been performed.

The following image is a List of Values (LOV) in Siebel CRM. Each LOV has one record per language. There is one record each for English-American and French languages. The Display Value displays in the user's selected language, while the Language-Independent Code (LIC) is the same for each pair. The LIC is often the same as the English Display Value because the Oracle engineers who originally created the LOV created everything in English and the Oracle translation team later created the translated records. However, this is not required. For example in the following image, you can see that the *English* "New Customer" value and *French* "Nouveau client" value share the same c LIC value.



Type	Display Value	Language-Independent Code	Language Name	Parent LIC	Order	Active	Translate	Multilingual	Replication Level
ACCOUNT_STATUS	Closed	Closed	English-American			☑	☑		All
ACCOUNT_STATUS	Fermé	Closed	French			☑	☑		All
ACCOUNT_STATUS	Actif	Active	French		3	☑	☑		All
ACCOUNT_STATUS	Active	Active	English-American		3	☑	☑		All
ACCOUNT_STATUS	New Customer	C	English-American		21	☑	☑		All
ACCOUNT_STATUS	Nouveau client	C	French		21	☑	☑		All

When the MLOV conversion has not occurred, picklists display all values from all languages. This is undesirable behavior since the user only wants to see values in their own language. The following image shows a Status picklist that displays values in both English and French (for example: Inactif, Red Customer, Client rouge, New Customer, Nouveau client, Other, Autre, and so on).



Account Name: A1 Manufacturing
Site: HQ
Account Team: SADMIN
Address: One Palmer Squ
State: NJ
Main Phone #: (908) 768-7867
City: Princeton
Country: USA
Main Fax #:
Zip Code: 08542
URL: www.a1man.com

Account Information

Account Name: A1 Manufacturing
Site: HQ

Pricing and Purchasing Information

Account Administration Details

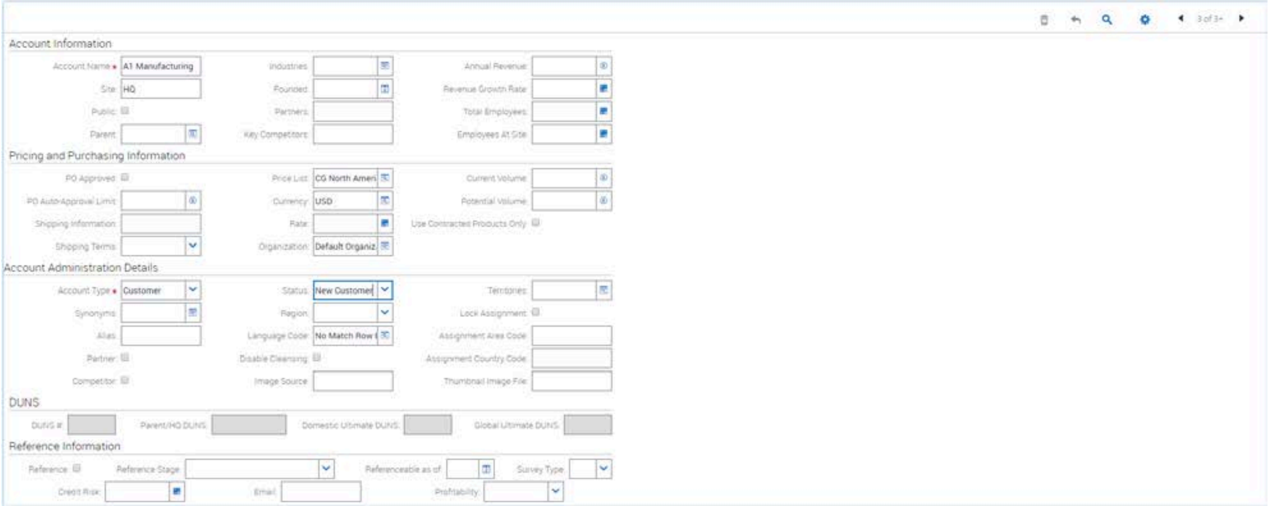
Account Type: Customer
Status: Active
Language Code: No Match Row I

DUNS

Reference Information

The Display Value is stored in the database and whatever value is stored there is visible to all users. Therefore, if an English-speaking user selects Active, the Active value is stored in the database. A French user who views that same record will also see Active (not Actif).

The following image displays what an English user sees when the New Customer value is selected – that is, an empty form with the following sections where the user can input data as required: Account Information, Pricing and Purchasing Information, Account Administration Details, DUNS, Reference Information.



Account Information

Account Name: A1 Manufacturing
Site: HQ

Pricing and Purchasing Information

Price List: CG North Ameri
Currency: USD
Organization: Default Organiz

Account Administration Details

Account Type: Customer
Status: New Customer
Language Code: No Match Row I

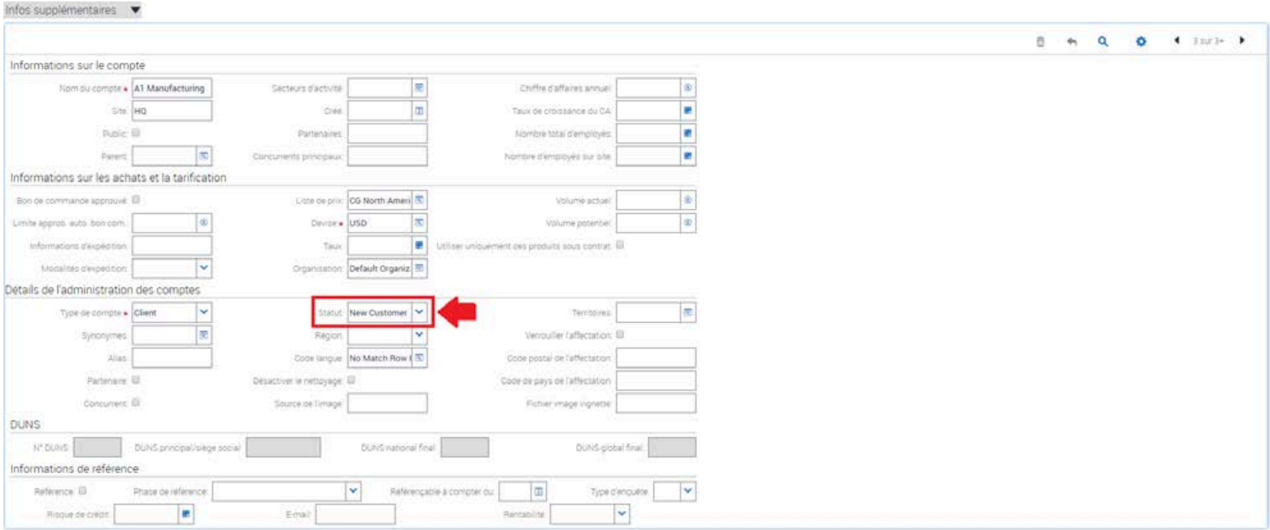
DUNS

Reference Information

Siebel CRM stores the New Customer value in the database as follows:

- Name: A1 Manufacturing
- CUST_STAT_CD: New Customer.

A French user viewing the same record sees the English version, as opposed to the desired French version of the record. The following image shows what a French user sees (when the New Customer value is selected) – in this image, the Statut field with the "New Customer" value is highlighted; the other sections on the form are: Informations sur le compte, Informations sur les achats et la tarification, Détails de l'administration de comptes, DUNS, Information de référence.



Informations sur le compte

Nom du compte *	A1 Manufacturing	Secteurs d'activité		Chiffre d'affaires annuel	
Site	HQ	Créé		Taux de croissance du CA	
Public	☐	Partenaires		Nombre total d'employés	
Parent		Concurrents principaux		Nombre d'employés sur site	

Informations sur les achats et la tarification

Bon de commande approuvé	☐	Liste de prix	CG North Ameri	Volume actuel	
Limite approb. auto. bon com.		Devise *	USD	Volume potentiel	
Informations d'expédition		Taux		Utiliser uniquement des produits sous contrat	☐
Modalités d'expédition		Organisation	Default Organiz		

Détails de l'administration des comptes

Type de compte *	Client	Statut	New Customer	Territoires	
Synonymes		Région		Verrouiller l'affectation	☐
Alias		Code langue	No Match Row I	Code postal de l'affectation	
Partenaire	☐	Désactiver le nettoyage	☐	Code de pays de l'affectation	
Concurrent	☐	Source de l'image		Fichier image vignette	

DUNS

N° DUNS		DUNS principal/siège social		DUNS national final		DUNS global final	

Informations de référence

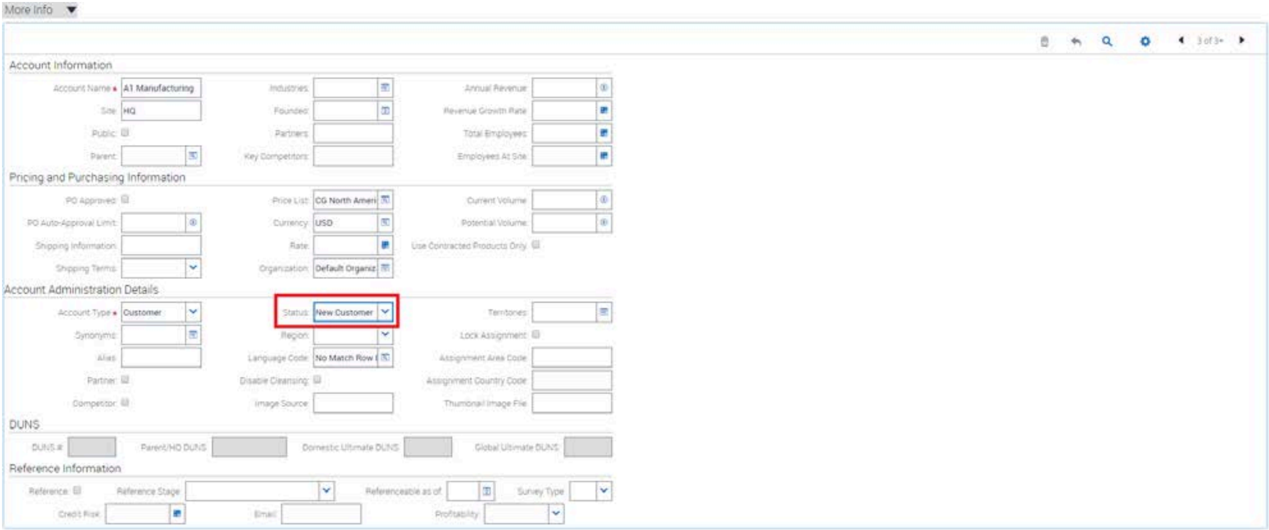
Référence	☐	Phase de référence		Référençable à compter du		Type d'enquête	
Risque de crédit		E-mail		Rentabilité			

The MLOV conversion process resolves these issues. The LOVs themselves are marked as Multilingual (the Multilingual check box is selected), indicating to the Siebel CRM platform that they must be treated as MLOVs and automatically translated.

During the MLOV conversion, existing records in the database are updated to contain the LIC value, rather than the language-specific Display Value. The record is stored with the LIC value in the database as follows:

- Name: A1 Manufacturing
- CUST_STAT_CD: C.

When a record is retrieved from the database, Siebel CRM automatically performs a mapping from the stored LIC values to the correct Display Value for the user's language. For example, the following image is what an English user sees after a MLOV conversion has been performed and the CUST_STAT_CD of C is stored in the database –– in this image, the Status field with the "New Customer" value is highlighted. The Status picklist will now show all options in English only.



Account Information

Account Name *	A1 Manufacturing	Industries		Annual Revenue	
Site	HQ	Founded		Revenue Growth Rate	
Public	☐	Partners		Total Employees	
Parent		Key Competitors		Employees At Site	

Pricing and Purchasing Information

PO Approved	☐	Price List	CG North Ameri	Current Volume	
PO Auto-Approval Limit		Currency	USD	Potential Volume	
Shipping Information		Rate		Use Contracted Products Only	☐
Shipping Terms		Organization	Default Organiz		

Account Administration Details

Account Type *	Customer	Status	New Customer	Territories	
Synonyms		Region		Lock Assignment	☐
Alias		Language Code	No Match Row I	Assignment Area Code	
Partner	☐	Disable Cleansing	☐	Assignment Country Code	
Competitor	☐	Image Source		Thumbnail Image File	

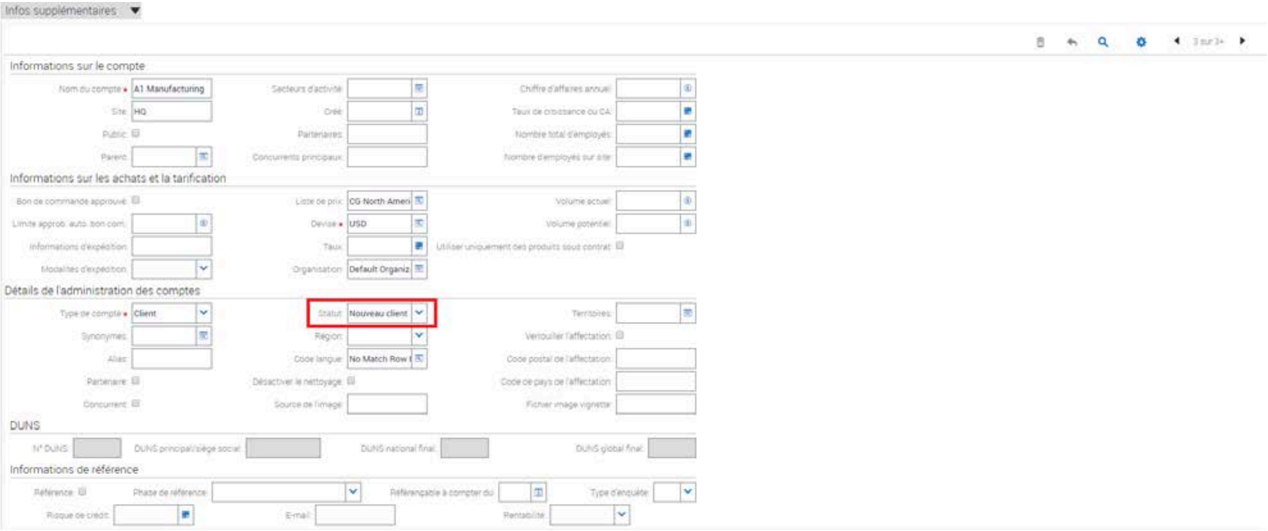
DUNS

DUNS #		Parent/HQ DUNS		Domestic Ultimate DUNS		Global Ultimate DUNS	

Reference Information

Reference	☐	Reference Stage		Referenceable as of		Survey Type	
Credit Risk		Email		Profitability			

The following image shows what a French user sees for the same record after the MLOV conversion process – in this image, the Statut field with the "Nouveau client" value is highlighted. The Statut picklist will show all options in French.



Informations sur le compte

Nom du compte *	A1 Manufacturing	Secteurs d'activité		Chiffre d'affaires annuel	
Site	HQ	Créé		Taux de croissance du CA	
Public	☐	Partenaires		Nombre total d'employés	
Parent		Concurrents principaux		Nombre d'employés sur site	

Informations sur les achats et la tarification

Bon de commande approuvé	☐	Liste de prix	CG North Ameri	Volume actuel	
Limite approb. auto. bon com.		Devise *	USD	Volume potentiel	
Informations d'expédition		Taux		Utiliser uniquement des produits sous contrat	☐
Modalités d'expédition		Organisation	Default Organiz		

Détails de l'administration des comptes

Type de compte *	Client	Statut	Nouveau client	Territoires	
Synonymes		Région		Verrouiller l'affectation	☐
Alias		Code langue	No Match Row I	Code postal de l'affectation	
Partenaire	☐	Désactiver le nettoyage	☐	Code de pays de l'affectation	
Concurrent	☐	Source de l'image		Fichier image vignette	

DUNS

N° DUNS		DUNS principal/siège social		DUNS national final		DUNS global final	

Informations de référence

Référence	☐	Phase de référence		Référençable à compter du		Type d'enquête	
Risque de crédit		E-mail		Rentabilité			

About Languages and Repository Data

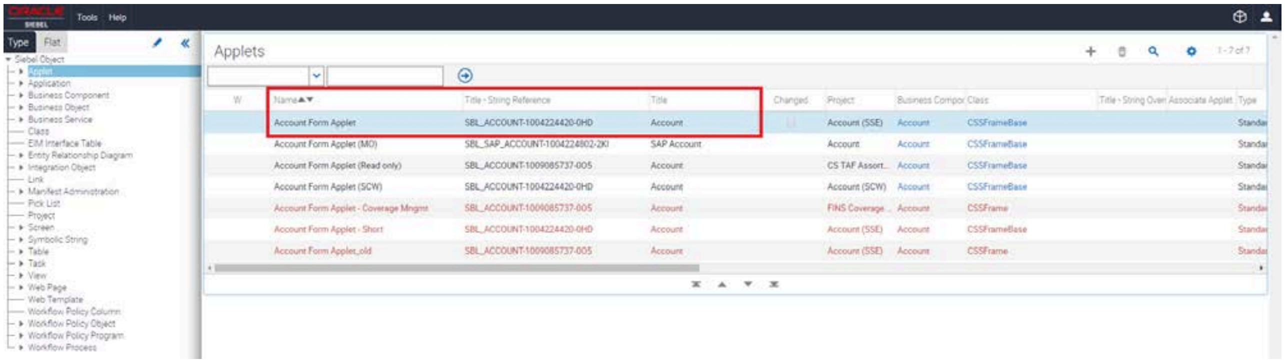
In addition to supporting multiple languages within Picklists, the Siebel CRM application supports displaying the user interface in multiple languages using the same repository. A brief description of how this works at the repository level is provided in this topic. For more information, see *Siebel Global Deployment Guide* .

From the point of view of globalization, repository data comes in two flavors: Language Neutral and Language Specific.

- Language neutral repository objects share the same definition across all languages. Generally speaking, they are those objects that have no user interface aspect to them. For example, a BusComp or a BusComp field definition is the same regardless of the language in which the user is working. The Account BusComp is always the Account BusComp, regardless of the user's language choice.
- Language specific objects need different definitions to support each user's language choice. For example, an English-speaking user would like to see an Account applet with the Accounts caption with columns such as Name or Location, while a French user would like to see that same applet rendered with *Comptes* as the caption and *Nom de compte* and *Site* columns.

To accommodate this behavior, many of the object types stored in the repository consist of parent object types that define the neutral components and child object types that contain the language specific aspects. Any attribute that can be translated, such as captions, labels, or column headers, are stored in the repository as symbolic strings, rather than hard-coded strings.

For example, consider the definition of the Account Form Applet in the following image where the Title-String Reference attribute value is `SBL_ACCOUNT-1004224420-0HD`.



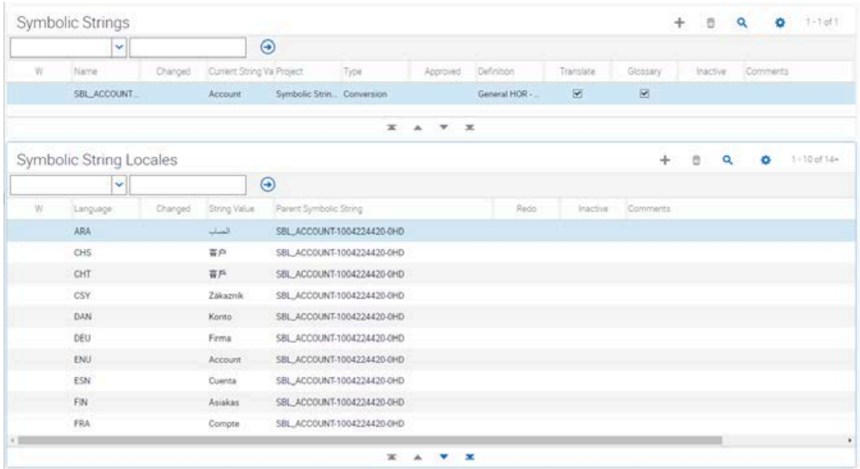
Applets

W	Name	Title - String Reference	Title	Changed	Project	Business Compor	Class	Title - String Overr	Associate Applet	Type
	Account Form Applet	SBL_ACCOUNT-1004224420-0HD	Account	☐	Account (SSE)	Account	CSSFrameBase			Standa
	Account Form Applet (MO)	SBL_SAP_ACCOUNT-1004224802-2KI	SAP Account		Account	Account	CSSFrameBase			Standa
	Account Form Applet (Read only)	SBL_ACCOUNT-1009085737-005	Account		CS TAF Assort...	Account	CSSFrameBase			Standa
	Account Form Applet (SCW)	SBL_ACCOUNT-1004224420-0HD	Account		Account (SCW)	Account	CSSFrameBase			Standa
	Account Form Applet - Coverage Mngmt	SBL_ACCOUNT-1009085737-005	Account		FINS Coverage ...	Account	CSSFrame			Standa
	Account Form Applet - Short	SBL_ACCOUNT-1004224420-0HD	Account		Account (SSE)	Account	CSSFrameBase			Standa
	Account Form Applet_old	SBL_ACCOUNT-1009085737-005	Account		Account (SSE)	Account	CSSFrame			Standa

This attribute value is used to look up the language-specific string at runtime based on the user's desired language.

Note: The Title field shows the value of the string for the current language. In this example, Siebel Web Tools is running in English, so it shows the English value, `Account`.

If you navigate to the symbolic string object and search for the `SBL_ACCOUNT-1004224420-0HD` attribute value, you will see a child record for each language. The following image shows the results of the `SBL_ACCOUNT-1004224420-0HD` attribute value search. In this image, a record exists for the following languages: ARA, CHS, DAN, DEU, ENU, ESN, FIN, FRA, ITA, and so on.



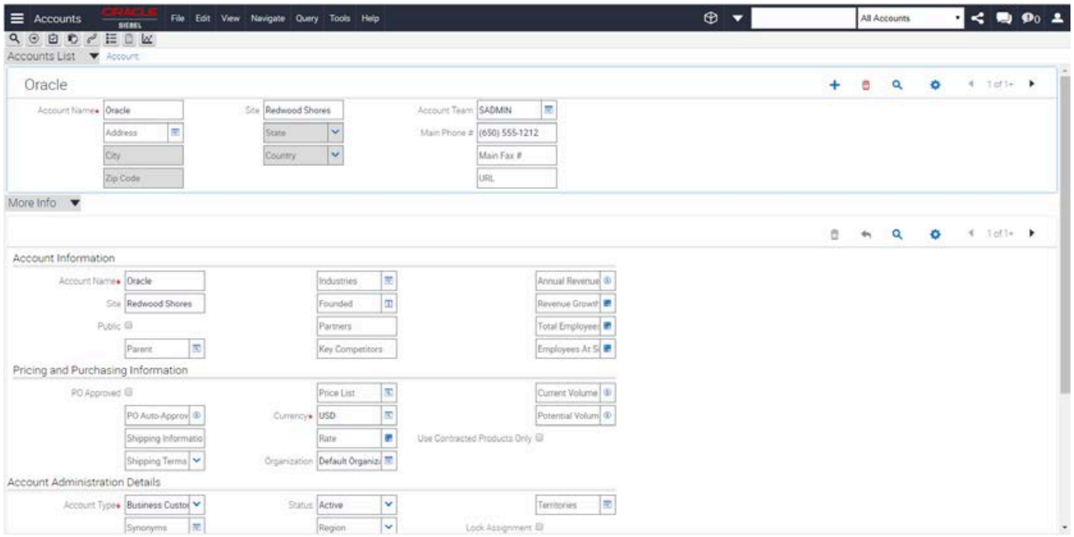
Symbolic Strings

W	Name	Changed	Current String Va	Project	Type	Approved	Definition	Translate	Glossary	Inactive	Comments
	SBL_ACCOUNT...		Account	Symbolic Strin...	Conversion		General HOR - ...	☑	☑		

Symbolic String Locales

W	Language	Changed	String Value	Parent Symbolic String	Redo	Inactive	Comments
	ARA		الحساب	SBL_ACCOUNT-1004224420-0HD			
	CHS		客户	SBL_ACCOUNT-1004224420-0HD			
	CHT		客戶	SBL_ACCOUNT-1004224420-0HD			
	CSY		Zákazník	SBL_ACCOUNT-1004224420-0HD			
	DAN		Konto	SBL_ACCOUNT-1004224420-0HD			
	DEU		Firma	SBL_ACCOUNT-1004224420-0HD			
	ENU		Account	SBL_ACCOUNT-1004224420-0HD			
	ESN		Cuenta	SBL_ACCOUNT-1004224420-0HD			
	FIN		Asiakas	SBL_ACCOUNT-1004224420-0HD			
	FRA		Compte	SBL_ACCOUNT-1004224420-0HD			

At runtime, the Siebel CRM application renders the applet using the symbolic string for the user's desired language. The following image shows the symbolic string when the user’s language is English. In this image, all field names and input values are in English.



Oracle

Account Name: Oracle
Site: Redwood Shores
Account Team: SADMIN
Main Phone #: (650) 555-1212

Account Name: Oracle
Site: Redwood Shores
Currency: USD
Organization: Default Organiz
Account Type: Business Custo
Status: Active

About the Process of Adding Additional Languages

To add additional languages to the Siebel CRM database, complete the following tasks:

1. *About Importing Languages*
   a. Import language-specific seed data.
   b. Import language-specific metadata to the Design Repository.
2. *About Flattening the Repository*
3. *About Running the Multilingual Conversion Process*
4. *About Publishing Changes*

> **Note:** It is not possible to add the binary components for additional languages after the initial installation. For example, if you install the SES and SAI components for English only, it will not be possible to add the French components later. You will need to uninstall and reinstall those components before adding new languages and running the MLOV conversion process. For more information, refer to Chapter 2, Installing Software for the Siebel CRM Application.

About Importing Languages

You can import individual languages by running the Database Server Configuration wizard for each language that you wish to install. You must run the Database Server Configuration wizard twice for each language. When you run the Database Server Configuration wizard, select the following options:

- **Install Database.** Select this option once for every language to import the seed data that supports each language. Seed data appears in picklists and elsewhere throughout the Siebel CRM application. Failure to run the installation process for each language will result in empty picklists for that language.

- **Import Repository.** Select this option once for every language to import the Siebel Runtime Repository data for each language. Siebel Runtime Repository data is the metadata that defines how the Siebel CRM application behaves. If you do not import the repository data for each language, the user interface will not show labels and other elements for that language.

About Flattening the Repository

After importing the seed and repository data for each additional language, the language data will exist in the database, but only seed data will be associated with the Workspace – the repository data will not be associated with the Workspace. This is illustrated by comparing ENU records to non-ENU records, as shown in the following tables.

The following table shows the Workspace-specified LOVs for newly imported languages.

NAME	VAL	LANG_ID	WS_ID
Active	Active	ENU	1@1
Active	Actif	FRA	1@1

The following table shows the symbolic string Repository data not specified for newly imported languages.

SYM_STR_KEY	LANG_CD	STRING_VALUE	REPOSITORY_ID	WS_ID
SBL_ACCOUNT-1004224420-0HD	ENU	Account	1–1–1	1@1
SBL_ACCOUNT-1004224420-0HD	FRA	Compte	1–1–1	

In order to set the Workspace Ids on the imported repository objects, you must flatten the repository. You can flatten the repository by running Siebel Tools with an appropriate set of command line parameters. For more information about flattening repositories, see *Using Siebel Tools* .

To flatten the repository, do the following:

1. In the tools.cfg file, set the following key to the ODBC source of your Siebel database:

```
[Siebel]
.....
ServerDbODBCDataSource = Siebel_DSN
...
```

2. From the command line, change to the Siebel Tools bin folder, and then execute the flatten database process by running Tools with the FlattenWorkspace command line argument:

   siebdev.exe /c tools.cfg /u UserID /p Password /d ServerDataSrc / FlattenWorkspace

After completing this process, all repository records for all languages are associated with the base Workspace, as shown by the sample record in the following table:

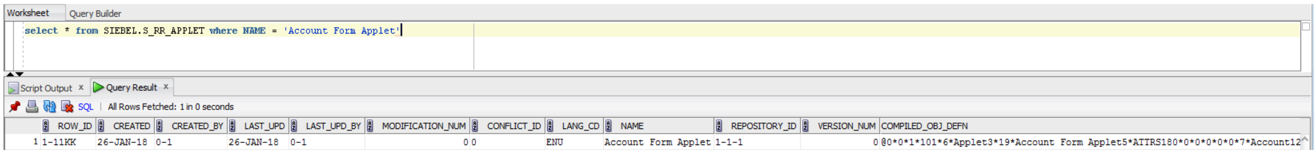
SYM_STR_KEY	LANG_CD	STRING_VALUE	REPOSITORY_ID	WS_ID
SBL_ACCOUNT-1004224420-0HD	ENU	Account	1–1–1	1@1
SBL_ACCOUNT-1004224420-0HD	FRA	Comte	1–1–1	**1@1**

About Running the Multilingual Conversion Process

The Multilingual Conversion process activates the multilingual list of values (MLOV) and ensures that the Siebel CRM application displays language-specific values in picklists throughout the Siebel CRM application. For more information about executing the Multilingual Conversion process, see *Siebel Global Deployment Guide* .

About Publishing Changes

The language import processes and the MLOV Conversion process makes changes to the Design Repository which are not automatically compiled into the Siebel Runtime Repository. The Siebel Runtime Repository requires a record for each language for each object. For example, an Account Form Applet has a record for each language. However, after importing languages, a Runtime Repository record exists only for the original language, typically English, as shown in the following image.



```
select * from SIEBEL.S_RR_APPLET where NAME = 'Account Form Applet'
```

All Rows Fetched: 1 in 0 seconds

```
   ROW_ID  CREATED    CREATED_BY  LAST_UPD   LAST_UPD_BY  MODIFICATION_NUM  CONFLICT_ID  LANG_CD  NAME                 REPOSITORY_ID  VERSION_NUM  COMPILED_OBJ_DEFN
 1 1-11KK  26-JAN-18  0-1         26-JAN-18  0-1                         0  0            ENU      Account Form Applet  1-1-1                    0  @0*0*1*101*6*Applet3*19*Account Form Applet5*ATTRS180*0*0*0*0*0*7*Account12
```

> **Note:** When the Runtime Repository is in this state, any attempt to login to Siebel CRM application returns a misleading error indicating Incorrect user ID or password. The actual issue is that the Siebel Runtime Repository and Workspace are in an invalid state. As a byproduct of the flattening process, even the base language users are unable to use the Siebel CRM application.

To address this problem and merge the changes and language-specific repository information into the Siebel Runtime Repository, you must perform a Full Publish. You can use the following command from the Siebel Tools bin folder to perform a Full Publish:

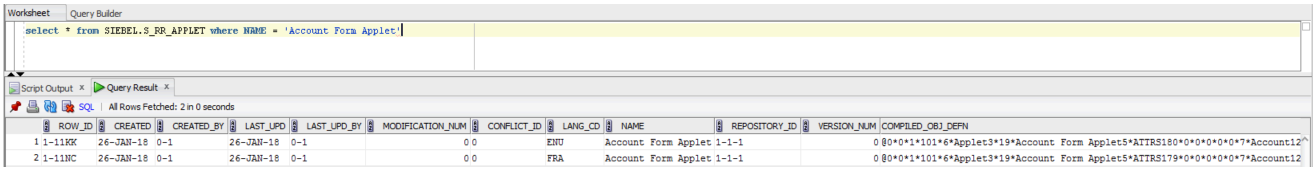
```
siebdev /c tools.cfg /TL {Comma-Separated Language Codes} /d ServerDataSrc /u SADMIN /p ******** /FullPublish
```

For example, the following command would regenerate the Runtime Repository for all 22 out-of-the-box languages:

```
siebdev /c tools.cfg /TL
ENU,ARA,CHS,CHT,CSY,DAN,DEU,ESN,FIN,FRA,HEB,ITA,JPN,KOR,NLD,PLK,PTB,PTG,RUS,SVE,THA,TRK /d
ServerDataSrc /u SADMIN /p ******* /FullPublish
```

> **Note:** At this time, Full Publish must be run on all desired languages at the same time. Processing occurs in parallel, which requires a lot of memory and requires a large number of database operations. For this reason, it is recommended that this process be run on a machine with a significant (16GB+) amount of memory and that the Full Publish process is run on a server on the same network subnet as the Siebel Database Server.

After the Full Publish process completes, the Siebel Runtime Repository is populated with the records supporting all languages. In the following image, since Full Publish was run for English and French, you can see that there are now entries in the Applet Runtime Repository table for both languages.



```
select * from SIEBEL.S_RR_APPLET where NAME = 'Account Form Applet'
```

All Rows Fetched: 2 in 0 seconds

	ROW_ID	CREATED	CREATED_BY	LAST_UPD	LAST_UPD_BY	MODIFICATION_NUM	CONFLICT_ID	LANG_CD	NAME	REPOSITORY_ID	VERSION_NUM	COMPILED_OBJ_DEFN
1	1-11KK	26-JAN-18	0-1	26-JAN-18	0-1	0	0	ENU	Account Form Applet	1-1-1	0	@0*0*1*101*6*Applet3*19*Account Form Applet5*ATTRS180*0*0*0*0*0*7*Account12
2	1-11NC	26-JAN-18	0-1	26-JAN-18	0-1	0	0	FRA	Account Form Applet	1-1-1	0	@0*0*1*101*6*Applet3*19*Account Form Applet5*ATTRS179*0*0*0*0*0*7*Account12