Siebel

Installation Guide

October 2026

Siebel
Installation Guide

October 2026

F84309-18

Copyright © 1994, 2026, Oracle and/or its affiliates.

Author: Sukanya Mishra, Carmen Myrick

Contents

Get Help i

1 What's New in This Release 1

What's New in Siebel Installation Guide, Siebel CRM 26.10 Update 1
What's New in Siebel Installation Guide, Siebel CRM 26.9 Update 1
What's New in Siebel Installation Guide, Siebel CRM 26.8 Update 2
What's New in Siebel Installation Guide, Siebel CRM 26.7 Update 2
What's New in Siebel Installation Guide, Siebel CRM 26.6 Update 2
What's New in Siebel Installation Guide, Siebel CRM 26.5 Update 3
What's New in Siebel Installation Guide, Siebel CRM 26.3 Update 3
What's New in Siebel Installation Guide, Siebel CRM 26.1 Update 4
What's New in Siebel Installation Guide, Siebel CRM 25.12 Update 4
What's New in Siebel Installation Guide, Siebel CRM 25.2 Update 4
What's New in Siebel Installation Guide, Siebel CRM 22.7 Update 5
What's New in Siebel Installation Guide, Siebel CRM 22.6 Update 5
What's New in Siebel Installation Guide, Siebel CRM 22.5 Update 5
What's New in Siebel Installation Guide, Siebel CRM 22.4 Update 6
What's New in Siebel Installation Guide, Siebel CRM 22.3 Update 6
What's New in Siebel Installation Guide, Siebel CRM 22.2 Update 6
What's New in Siebel Installation Guide, Siebel CRM 22.1 Update 7
What's New in Siebel Installation Guide, Siebel CRM 21.12 Update 7
What's New in Siebel Installation Guide, Siebel CRM 21.11 Update 7
What's New in Siebel Installation Guide, Siebel CRM 21.10 Update 8
What's New in Siebel Installation Guide, Siebel CRM 21.9 Update 8
What's New in Siebel Installation Guide, Siebel CRM 21.8 Update 9
What's New in Siebel Installation Guide, Siebel CRM 21.7 Update 9
What's New in Siebel Installation Guide, Siebel CRM 21.6 Update 10
What's New in Siebel Installation Guide, Siebel CRM 21.5 Update 10
What's New in Siebel Installation Guide, Siebel CRM 21.4 Update 11
What's New in Siebel Installation Guide, Siebel CRM 21.3 Update 12
What's New in Siebel Installation Guide, Siebel CRM 21.2 Update 12
Additional Information 13

2 Overview of Installing Siebel CRM 15

Overview of Installing Siebel CRM 15
Installation-Related Changes for Siebel CRM 15
Before You Install Siebel CRM 21
General Guidelines for Installing Siebel CRM 22
Overview of Siebel CRM Server Architecture 24
Roadmap: Installing Siebel CRM for a New Deployment 25
Roadmap: Installing Siebel CRM in an Update Installation Case (Existing Installation of Siebel CRM 17.x or Later) 28
Roadmap: Installing Siebel CRM in a Migration Installation Case (Existing Installation of Siebel CRM 16.x or Earlier) 29
Roadmap: Installing Siebel CRM in an Upgrade Installation Case (Existing Database Requiring Full Upgrade) 33

3 Siebel CRM Download and Installation 37

Siebel CRM Download and Installation 37
Finding and Downloading a Siebel CRM Release/Update Image Content 37
About Installation Tasks for Siebel CRM 43
Installing Siebel CRM 44
Installing Siebel CRM Using Silent Mode 62
Additional Tasks for Migration Installations 70

4 Configuring Siebel CRM Server Modules 75

Configuring Siebel CRM Server Modules 75
About Configuring Siebel CRM 75
Running the Siebel Management Console 79
Installing the Siebel Database 87
Configuring the Siebel Gateway and Security 87
Configuring the Siebel Enterprise 96
Configuring the Siebel Server 100
Configuring the Siebel Application Interface 105
Configuring the Siebel Gateway Cluster 116
Configuring the Siebel Migration Application 124
Configuring the Siebel Deployment 130
Managing the Siebel Deployment 133
Accessibility for Siebel Management Console 136

5 Installing the Siebel Database on the RDBMS 143

Installing the Siebel Database on the RDBMS 143
About Database Updates for Siebel CRM 143
About Installing the Siebel Database 145
Creating Table Owner and Administrator Accounts 147
Installing the Stored Procedures and User-Defined Functions on IBM DB2 151
Starting the Siebel Database Configuration Wizard 152
Installing the Siebel Database 156
Activating License Keys 168
Setting Up Your Environment to Support Global Time Zone 169
Populating the Siebel File System 170
Importing a Siebel Repository Manually into the Siebel Database 170

6 Additional Postinstallation and Configuration Tasks 175

Additional Postinstallation and Configuration Tasks 175
Importing Siebns.dat into Zookeeper 175
Starting the Siebel Gateway 176
Stopping and Starting the Siebel Application Container 178
Configuring Siebel Application Containers for Automatic Start on UNIX 180
Starting the Siebel Server Service on Windows 182
Configuring the Siebel Gateway for Automatic Start on UNIX 182
Configuring the Siebel Server for Automatic Start on UNIX 186
Managing Environment Variables on UNIX 188
Preparing to Run Siebel Server Components 190
Customizing the Application Container for Siebel Application Interface 194
Installing and Deploying Siebel CRM with Multiple Languages 201
Configuring Support for CalDAV or CardDAV Publishing 210
Inactivating Workflow Processes 211

7 Installing Siebel Web Clients and Siebel Tools 213

Installing Siebel Web Clients and Siebel Tools 213
About Installing Siebel Web Client or Siebel Tools 213
Installing and Using Oracle Database SE2 for the Local Database 215
About Workspaces in Siebel Tools and the Siebel Runtime Repository 218
Installing Desktop Integration Siebel Agent on Client Computers 219
Siebel Client Deployment Requirements When Using Firewalls or Proxy Servers 219
Configuring Siebel CRM for Pop-Up Blocker Compatibility 220

Logging in to Your Siebel Application 221
Siebel Web Client Shortcuts and Startup Options 223

8 Verifying and Troubleshooting Your Installation 227
Verifying and Troubleshooting Your Installation 227
Verifying Installation for Siebel CRM 227
Troubleshooting Installation and Configuration for Siebel CRM 237
Verifying the ODBC Data Source 241
Troubleshooting the ODBC Data Source Connection 245
Verifying Installation for the Siebel Database 246
Troubleshooting Installation for the Siebel Database 249
Verifying Installation for Siebel Tools 251
Verifying Installation for the Siebel Web Client 254
Verifying Installation for Oracle Database SE2 256
Verifying Network Connectivity for the Siebel Server Computer 258
Verifying Network Connectivity for Mobile Users 259
Configuring the Siebel Mobile Web Client When Neither Siebel VB nor Siebel eScript Is Licensed 259

9 Uninstalling Siebel CRM 261
Uninstalling Siebel CRM 261
About Uninstalling Siebel CRM 261
Process of Removing Configuration Data 265
Uninstalling Siebel CRM 268
Uninstalling Oracle Database XE 272
About Uninstalling Earlier Versions of Siebel CRM 273

10 Requirements for Installing and Configuring Siebel CRM 277
Requirements for Installing and Configuring Siebel CRM 277
About Installing Siebel CRM 277
Planning Your Siebel Deployment 280
Requirements for Installing and Configuring Siebel Enterprise Components 301
Requirements for Installing and Configuring the Siebel Application Interface 308
Requirements for Installing and Configuring the Siebel Database 314
Requirements for Installing Siebel Web Clients 316
Requirements for Installing Siebel Tools 319

11 Configuring the RDBMS 321
Configuring the RDBMS 321
Overview of Database Configuration 321
Configuring an Oracle Database for Siebel CRM 323
Configuring an IBM DB2 Database for Siebel CRM 331
Configuring a Microsoft SQL Server Database for Siebel CRM 344
Configuring Siebel Server Connectivity to the Siebel Database 349

12 Siebel Deployment Planning Worksheet 351
Siebel Deployment Planning Worksheet 351
About the Deployment Planning Worksheet 351
Team Lead Summary 352
Siebel Enterprise Server Names and Installation Directories 353
Siebel Accounts, Host Names, and Static IP Addresses 354
Cluster Deployment Data 355
Ports and RDBMS Details Data 356

13 Siebel Open Integration 359
Siebel Open Integration 359
Installing Siebel Open Integration 359
Configuring and Deploying Siebel Open Integration 362
Design Time Workflow and Supporting Artifacts 420
Managing Security and Authentication 421
Configuring Open Integration Enterprise Cache for Siebel 429
Siebel AI Connectors 436
Configure Open Integration Logging 472

Get Help

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 *https://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 Siebel Installation Guide, Siebel CRM 26.10 Update

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

Topic	Description
Configure Open Integration Logging	Modified topic. This topic describes the how you can configure Open Integration logging.

What's New in Siebel Installation Guide, Siebel CRM 26.9 Update

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

Topic	Description
Securing Siebel Management Console Deployments	New topic. It describes how to securely deploy and control access to Siebel Management Console (SMC) and Application Interface nodes.
Configuring Open Integration Logging	New topic. Describes how to configure Open Integration application and dead-letter queue (DLQ) logging, including log locations, console-only logging, supported log levels, environment variables, and `log4j2.xml` settings.

What's New in Siebel Installation Guide, Siebel CRM 26.8 Update

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

Topic	Description
Configuring Open Integration Enterprise Cache for Siebel	New topic. This topic describes how you can configure Enterprise Cache for Open Integration in SMC.

What's New in Siebel Installation Guide, Siebel CRM 26.7 Update

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

Topic	Description
Siebel AI Connectors	Added topic. Provides an overview of Siebel AI Connectors.

What's New in Siebel Installation Guide, Siebel CRM 26.6 Update

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

Topic	Description
Configuring Siebel Open Integration	Modified topic. Open Integration introduces an enhanced integration framework that enables organizations to implement advanced integration mechanisms for seamless communication between systems.

What's New in Siebel Installation Guide, Siebel CRM 26.5 Update

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

Topic	Description
Siebel Open Integration	Modified the Siebel Open Integration topic in the chapter 13 Siebel Open Integration.
Design Time Workflow and Supporting Artifacts	Modified the Design-Time Workflow and Supporting Artifacts topic in the chapter 13 Siebel Open Integration.
Managing Security and Authentication	Modified the Managing Security and Authentication topic in the chapter 13 Siebel Open Integration.
Configuring Advanced Cache in SMC for On Premises Open Integration	Modified the Configuring Advanced Cache in SMC for On-Premises Open Integration topic in the chapter 13 Siebel Open Integration.

What's New in Siebel Installation Guide, Siebel CRM 26.3 Update

Topic	Description
Installing Siebel Open Integration on Windows	New topic. This topic describes how to deploy Siebel Open Integration on Windows in interactive mode and silent mode.
Installing Siebel Open Integration on Linux	New topic. This topic describes how to deploy Siebel Open Integration on Linux in interactive mode and silent mode.

What's New in Siebel Installation Guide, Siebel CRM 26.1 Update

Topic	Description
Running the grantusr.sql scripts on Azure SQL	New topic. This topic explains how to run the grantusr_azure_master.sql and grantusr_azure_siebeldb.sql scripts to set up necessary logins and users for Siebel CRM on Azure SQL. .
Multiple topics across the guide.	This guide is updated to replace *RepositoryUpgrade* with *RepositoryUpdate*. This updated process minimizes the number of changes applied by the utility, resulting in a smoother adoption of content introduced in Siebel Monthly Updates and enabling customers to more easily leverage new features.

What's New in Siebel Installation Guide, Siebel CRM 25.12 Update

Topic	Description
Siebel Open Integration	New chapter. This chapter provides an overview of how Siebel Open Integration can be installed. It explains how Open Integration can be configured with on-premises or cloud applications.

What's New in Siebel Installation Guide, Siebel CRM 25.2 Update

Topic	Description
Installation-Related Changes in Siebel CRM	Revised download and image creation content in *Siebel CRM Download and Installation* Removed legacy information. General cleanup of old information.

What's New in Siebel Installation Guide, Siebel CRM 22.7 Update

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

Topic	Description
Installation-Related Changes in Siebel CRM	Modified topic. Updated topics for Siebel CRM 22.7. **Note:** For the purposes of this documentation, the *current release* is Siebel CRM 22.7. (In this guide, the terms Siebel CRM 22.x and Siebel CRM 22.x Update mean the same thing.) **Note:** This guide might not be updated in every subsequent monthly release of Siebel CRM 22.x. In this case, use the latest available installation guide for your subsequent Siebel CRM 22.x release and substitute your version number as appropriate.
About the Migration Package Location	Modified topic. The Siebel Application Interface owner account must have read-write access to the Siebel File System and the Migration Package Location.

What's New in Siebel Installation Guide, Siebel CRM 22.6 Update

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

Topic	Description
Installation-Related Changes in Siebel CRM	Modified topic. Updated topics for Siebel CRM 22.6.

What's New in Siebel Installation Guide, Siebel CRM 22.5 Update

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

Topic	Description
Installation-Related Changes in Siebel CRM	Modified topic. Updated topics for Siebel CRM 22.5.

What's New in Siebel Installation Guide, Siebel CRM 22.4 Update

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

Topic	Description
Installation-Related Changes in Siebel CRM	Modified topic. Updated topics for Siebel CRM 22.4.

What's New in Siebel Installation Guide, Siebel CRM 22.3 Update

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

Topic	Description
Installation-Related Changes in Siebel CRM	Modified topic. Updated topics for Siebel CRM 22.3.

What's New in Siebel Installation Guide, Siebel CRM 22.2 Update

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

Topic	Description
Installation-Related Changes in Siebel CRM	Modified topic. Updated topics for Siebel CRM 22.2.
Reviewing the Installation Directories for Siebel Enterprise Components (and following topics)	Modified topics. Updated information about Siebel CRM directories for UNIX.

Topic	Description

What's New in Siebel Installation Guide, Siebel CRM 22.1 Update

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

Topic	Description
Installation-Related Changes in Siebel CRM	Modified topic. Updated topics for Siebel CRM 22.1.

What's New in Siebel Installation Guide, Siebel CRM 21.12 Update

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

Topic	Description
Installation-Related Changes in Siebel CRM	Modified topic. Updated topics for Siebel CRM 21.12.
Configuration Settings for a Siebel Application Interface Profile	Modified topic. As of Siebel CRM 21.12, for the OAuth authentication type for REST inbound authentication, the Client Id and Client Secret settings are new, and the Authentication URL setting has been renamed to be Introspection URL. For more information, see *Siebel REST API Guide* .
Multiple topics (modified) Configuring Siebel Enterprise Cache (removed)	Modified and removed topics. Siebel Enterprise Cache is no longer supported for Siebel Product Configurator (same as with Siebel Constraint Engine) and is not installed or updated by the new installer. Configuration options for Enterprise Cache have been removed from Siebel Management Console.

What's New in Siebel Installation Guide, Siebel CRM 21.11 Update

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

Topic	Description
Installation-Related Changes in Siebel CRM	Modified topic. Updated topics for Siebel CRM 21.11.
Installing Siebel CRM in an Update Installation *General Requirements for Installing and Configuring the Siebel Application Interface*	Modified topic. In some update installation cases, you must update Siebel Application Interface before you update the other installed modules.

What's New in Siebel Installation Guide, Siebel CRM 21.10 Update

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

Topic	Description
Installation-Related Changes in Siebel CRM	Modified topic. Updated topics for Siebel CRM 21.10.

What's New in Siebel Installation Guide, Siebel CRM 21.9 Update

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

Topic	Description
Installation-Related Changes in Siebel CRM	Modified topic. Updated topics for Siebel CRM 21.9.
Configuring Siebel CRM Using Silent Mode	Modified topic. Added an example of modifying the GatewaySecurityProfile.json file for database authentication.
Guidelines for Optimizing Oracle Parallel Queries for Siebel Web Tools	New topic. As of Siebel CRM 21.3, a new Siebel component parameter, Oracle Degree of Parallelism (alias OraDegreeOfParallelism), has been added, which allows setting the degree of parallelism for Siebel Web Tools or Siebel Tools when used with an Oracle Database.

What's New in Siebel Installation Guide, Siebel CRM 21.8 Update

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

Topic	Description
Installation-Related Changes in Siebel CRM	Modified topic. Updated topics for Siebel CRM 21.8.

What's New in Siebel Installation Guide, Siebel CRM 21.7 Update

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

Topic	Description
Installation-Related Changes in Siebel CRM	Modified topic. Updated topics for Siebel CRM 21.7.
Installation-Related Changes in Siebel CRM	Modified topic. Removed the incorrect statement that it is not necessary to include the non-ENU safeboot media in the Siebel image for subsequent installations of later releases. Supporting any non-ENU language requires that this media be part of the Siebel image before you install any Siebel CRM release.
Installing Siebel CRM in an Update Installation *Restoring a Prior Installation and Configuration*	Modified topics. It is strongly recommended to back up the Siebel Gateway registry before installing Siebel CRM as an update installation. After installing Siebel CRM, if you decide to roll back the installation and restore your prior release, then, as part of this task, you can use this backup to restore the Siebel Gateway registry to its prior state.
Performing an Installation in Silent Mode	Modified topic. The commands for performing silent installations have been updated to include the required flags for specifying the database table owner and database user password.
Creating the Siebel Service Owner Account	Modified topic. The Siebel service owner account name must not exceed 30 characters in length, including domain information..

What's New in Siebel Installation Guide, Siebel CRM 21.6 Update

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

Topic	Description
Installation-Related Changes in Siebel CRM	Modified topic. Updated topics for Siebel CRM 21.6.
Installing Siebel CRM in an Update Installation *Restoring a Prior Installation and Configuration*	Modified topics. It is strongly recommended to back up the Siebel database before installing Siebel CRM as an update installation (in which the installer runs PostInstallDBSetup). After installing Siebel CRM, if you decide to roll back the installation and restore your prior release, then, as part of this task, you can use this backup to restore the Siebel database to its prior state.
About the Migration Package Location *Creating the Siebel Service Owner Account*	Modified topics. The Siebel service owner account must have read-write access to the Siebel File System and the Migration Package Location.
Managing Environment Variables on UNIX	Modified topic. Added mention of the RESOLV_MULTI environment variable, which is required for Enterprise deployment on Linux.
Configuring the Common Logger for Siebel Application Interface	Modified topic. Moved most of the information in this topic into *Siebel System Monitoring and Diagnostics Guide* .

What's New in Siebel Installation Guide, Siebel CRM 21.5 Update

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

Topic	Description
Installation-Related Changes in Siebel CRM	Modified topic. Updated topics for Siebel CRM 21.5.
Installation-Related Changes in Siebel CRM	Modified topic. Removed the incorrect statement that U.S. English (ENU) is always installed.
Installing Siebel CRM in a New Installation	Modified topic. Updated the new installation procedure to include the Application Context Name field, which is part of the Application Interface Authentication screen, as of Siebel CRM 21.2.

Topic	Description
	Updated the new installation procedure to include installing and configuring Developer Web Client, which is supported by the Siebel CRM installer as of Siebel CRM 21.5.
Customizing URLs for Siebel CRM Applications	Modified topic. Updated procedure to accommodate product changes related to the new installer in Siebel CRM 21.2 and later.
Restoring a Prior Installation and Configuration	Modified topic. Provided more specific steps for restoring a prior installation that does not use the new installer (Siebel CRM 21.1 or earlier).

What's New in Siebel Installation Guide, Siebel CRM 21.4 Update

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

Topic	Description
Installation-Related Changes in Siebel CRM	Modified topic. Updated topics for Siebel CRM 21.4.
Installation-Related Changes in Siebel CRM	Modified topic. Updated information about installing Oracle Database Client for Siebel Web Client or for Siebel Server.
Installing Siebel CRM in a New Installation *Installing Siebel CRM in a Migration Installation*	Modified topics. As of Siebel CRM 21.4, for Siebel Application Interface new and migration installations, the default port number for HTTP Connection Port changed from 8080 to 80.
About Database Updates for Siebel CRM	Modified topic. Updated information about the PostInstallDBSetup utility. As of Siebel CRM 21.4, a menu is provided to specify options for running the utility.
Reviewing the Installation Directories for Siebel Gateway	Modified topic. As of Siebel CRM 21.4, an updated version of Apache ZooKeeper is provided for the Siebel Gateway registry. After installation of Siebel CRM 21.4 or later, note the following changes in the Siebel Gateway installation, located in `SIEBEL_ROOT\gtwysrvr`: • The `zookeeper` directory has been renamed as the `registry` directory. • The zoo1.cfg file has been renamed as the registry.cfg file. This file, formerly located in the `zookeeper\conf` directory, is now located in `registry\conf`. • The `version-2` directory, formerly located in the `zookeeper` directory, is now located in `registry\conf`. **Note:** The change to a new version of ZooKeeper does not require any customer backup or restore steps. Do not restore a backup of the Siebel Gateway registry from the prior version of ZooKeeper into the new version. For more information, see *Siebel System Administration Guide* .

What's New in Siebel Installation Guide, Siebel CRM 21.3 Update

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

Topic	Description
Installation-Related Changes in Siebel CRM	Modified topic. Updated topics for Siebel CRM 21.3.
Configuration Settings for a Siebel Application Interface Profile	Modified topic. As of Siebel CRM 21.3, the default value of the HTTP 1.1-Compliant Firewall / Enable Web Compression setting has changed to be enabled by default. This setting applies only to dynamic content.
About Database Updates for Siebel CRM	Modified topic. Updated information about the RepositoryUpgrade utility. As of Siebel CRM 21.3, PostInstallDBSetup no longer runs RepositoryUpgrade.
Compressing Static Content for Siebel Application Interface	New topic. You can now specify HTML compression for static content for Siebel Application Interface.
Load Balancing Siebel Application Interface by Configuring the workers.properties File	New topic. Moved topic here from *Siebel REST API Guide* .

What's New in Siebel Installation Guide, Siebel CRM 21.2 Update

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

Topic	Description
Installation-Related Changes in Siebel CRM	Modified topic. Updated topics for Siebel CRM 21.2. This guide describes using the new Siebel CRM installer, which is new as of Siebel CRM 21.2. This installer replaces the installers that were provided for prior releases. For new or migration installation cases, you no longer need to install or migrate a base release of a previous version. This new guide replaces the guides that were provided for Siebel CRM 21.1 or earlier and provides the relevant information that applies to all supported operating systems and RDBMS platforms.

Additional Information

For more information about Siebel CRM releases, see *Siebel CRM Update Guide and Release Notes* on My Oracle Support, KB825124, for each applicable release.

Platform, database, and other certifications for Siebel are found on My Oracle Support. For help searching for certifications, see to *KB871388*.

Siebel CRM makes extensive use of application containers. For detailed information about the configuration and administration of an application container outside of the Siebel configuration framework, see the Apache Tomcat documentation. See also *Stopping and Starting the Siebel Application Container* and *Customizing the Application Container for Siebel Application Interface*.

2 Overview of Installing Siebel CRM

Overview of Installing Siebel CRM

This chapter provides an overview of installing Siebel CRM. It includes the following topics:

- *Installation-Related Changes in Siebel CRM*
- *Before You Install Siebel CRM*
- *General Guidelines for Installing Siebel CRM*
- *Overview of Siebel CRM Server Architecture*
- *Roadmap: Installing Siebel CRM for a New Deployment*
- *Roadmap: Installing Siebel CRM in an Update Installation Case (Existing Installation of Siebel CRM 25.x or Later)*
- *Roadmap: Installing Siebel CRM in a Migration Installation Case (Existing Installation of Siebel CRM 16.x or Earlier)*
- *Roadmap: Installing Siebel CRM in an Upgrade Case (Existing Database Requiring Full Upgrade)*

Installation-Related Changes for Siebel CRM

Documentation Improvements for Siebel CRM

This guide describes using the Siebel CRM installer for releases at or above this release version unless a newer bookshelf has been made available.

This guide will be updated again if the deployment requirements change for later Siebel CRM releases.

The following improvements in Siebel CRM business agility, application deployment and test automation it possible to move to a new release model. The new model provides the ability to deliver new features and innovation faster to our customers. Customers that are on our current release can now embrace frequent innovation updates without the need for major upgrades.

Here are the key benefits:

- All future features and fixes to be delivered in “Updates”.
- The updates can contain features as identified from the latest Statement of Direction. For example, updates containing features, coming in 2025 will relate to the "Siebel CRM 2025 Statement of Direction".
- We are rationalizing the naming process to be more logical and related to the date of release, so an update in April 2025 will simply be called 25.4 applying an update, is not an upgrade (no IRM process or repository changes). Certain updates in the future that contain particular types of features may require IRM to be used. Updates are cumulative.
- This new model means customers on 25.x and above will never run out of updates for their release.
- This greatly simplifies the application update process, reducing maintenance costs over the old application upgrade approach.

- There are no major changes for customers on Siebel CRM 25 and above, other than the fact that customers can look forward to a more continuous delivery in the future.
- Alignment with agile development and deliver cumulative releases.

Updates and Changes

Release updates for Siebel CRM new features and bug fixes are released as 'Updates'.

We recommend that customers stay current with updates. By doing this they minimize the chance of encountering known bugs and security vulnerabilities. Staying current with Updates reduces the likelihood of requiring separate interim one-off patch fixes which lead to increased maintenance.

Lifetime Support Policy for Siebel CRM

To review the support life of Siebel CRM releases, please refer to the Lifetime Support Policy document.

Release Version Numbering Changes

The current notation for the Siebel CRM software updates is Year. Update is used, such as 25.2. This allows clear indication of the year and month of the update.

Installation Cases

Note the characteristics of the installation cases supported by this installer. More information is provided later in this topic. For new or migration installation cases, you no longer need to install or migrate a base release of a previous version. Existing requirements still apply for installing or upgrading the Siebel database, running Siebel Management Console, and so on. Roadmap topics are provided to guide your actions in each of the installation cases.

- **New installation.** For a new installation, you install Siebel CRM directly. See *Roadmap: Installing Siebel CRM for a New Deployment*.
- **Update installation for existing installation of Siebel CRM 17.x or later.** If you have an existing installation of Siebel CRM 17.x or later, then you install the current release of Siebel CRM to update your existing installation. See *Roadmap: Installing Siebel CRM in an Update Installation Case (Existing Installation of Siebel CRM 25.x or Later)*.
- **Migration installation for existing installation of Siebel CRM 16.x or earlier.** If you have an existing installation of a release prior to Siebel CRM 17.0 (Siebel CRM 8.1.1 or 8.2.2 through 16.x), then you can install Siebel CRM as a migration installation. Afterward, run Incremental Repository Merge (IRM). See *Roadmap: Installing Siebel CRM in a Migration Installation Case (Existing Installation of Siebel CRM 16.x or Earlier)*.
- **Installation in an upgrade case.** Where a full database upgrade must be performed, you can perform either new or migration installations, before you perform the database upgrade tasks. See also *Roadmap: Installing Siebel CRM in an Upgrade Case (Existing Database Requiring Full Upgrade)*. In general, this roadmap assumes that you are performing new installations.

In release media or software contexts showing the Siebel version number, the version is often represented as version 25.x, such as version 25.2 for Siebel CRM 25.2. When you install a subsequent Siebel CRM release, the version number of the installed software increments to version 25.3, 25.4, and so on, according to the release installed.

Siebel Database Updates for Update Installations

When you install Siebel CRM to update an existing installation of Siebel CRM that includes Database Configuration Utilities, after installation, the installer runs the Post Installation Database Update (PostInstallDBSetup) utility on the

Siebel database. PostInstallDBSetup runs several processes to ensure that the customer database schema, manifest, and seed data are up to date for the current monthly update release. (Database Configuration Utilities is no longer a selectable option for a new installation, but is always installed with Siebel Enterprise Components.) Where applicable, you can also optionally run the RepositoryUpdate utility to support new features.

See also the information later in this topic about the new database repository and ancestor database repository support files. For more information about the Siebel database, see the following:

- *About Database Updates for Siebel CRM* and other relevant content in this guide.
- *Siebel CRM Update Guide and Release Notes* on My Oracle Support for the latest Siebel CRM release.
- For supported Siebel database upgrade paths and instructions, see *Siebel Database Upgrade Guide* .

Product Media

Media packaging has been optimized and reorganized. You obtain the Siebel CRM product media as ZIP files that require no special utility to prepare for use. Siebel Image Creator utility is no longer provided.

Media are reorganized relative to previous releases and include modules not previously part of Siebel CRM media or installation. For example:

- For all installation cases, only Siebel CRM media of the version installed are required.
- Media are consolidated for server and client modules (for Microsoft Windows).
- No separate language media are provided, apart from the ZIP files for installing non-ENU safeboot files.
- Media for new database repository and ancestor database repository support files are provided for inclusion in Siebel CRM server installations, where required for your deployment.
- Media for Oracle Database SE2 are provided for inclusion in Siebel Web Client installations (for Siebel Mobile Web Client).
- Media for Oracle Database Client is automatically installed for both server and client. Starting with Siebel CRM 25.2. The database client is updated via Siebel updates only.

Installation Characteristics

Note the following installation characteristics for Siebel CRM:

- Each installation of Siebel CRM contains a complete set of binary files. Installations of Siebel CRM no longer require first creating or migrating a base installation using the installer for Siebel CRM 17.0.
- The installations that you perform are faster and in many cases can be performed in a single installation session.
- Installations require significantly less disk space. Duplication of language-related files and binary files across multiple components has been reduced or eliminated. For example:
  - (Windows) `SIEBEL_ROOT\gtwysrvr\bin`, `SIEBEL_ROOT\dbsrvr\bin`, and `SIEBEL_ROOT\eaiconn\bin` are now symbolic links to `SIEBEL_ROOT\siebsrvr\bin`.
  - (LINUX) `SIEBEL_ROOT/gtwysrvr/lib`, `SIEBEL_ROOT/dbsrvr/lib`, and `SIEBEL_ROOT/eaiconn/lib` are now symbolic links to `SIEBEL_ROOT/siebsrvr/lib`.
- For all server installations (new, update, or migration installation cases), binaries for all Siebel Enterprise components are included. Individual selections for Siebel Server, Siebel Gateway, Siebel Application Interface, EAI Connector, Database Configuration Utilities, and other components have been removed, because these modules or their equivalents are always installed for Siebel Enterprise Components. Siebel Management Console configuration tasks are not affected by these installation changes. Newly added components are

available to be configured if you choose. You must keep track of the modules you are configuring and deploying. Guidelines for nonproduction and production environments are provided here.

Some components could not be collocated in prior releases, but such limitations no longer apply and the components are now installed together. Note, however, that if you collocate modules (that is, install and deploy them together), you must consider various issues described in *Siebel Deployment Planning Guide* , including those related to the application container, which is shared when Siebel Server and Siebel Gateway are collocated. Siebel Application Interface uses its own application container even where it is collocated with Siebel Server or Siebel Gateway. Customers must consider such factors, among others, when planning their Siebel CRM deployment topology and before installing and deploying Siebel CRM server modules.

- Make installation selections that are consistent with your expected topologies or deployment environments, based in part on your security requirements. See also the information about client installation later in this topic. For example:
  - **Development or test environments (nonproduction).** You can install Siebel Enterprise Components and other modules, optionally including Siebel Web Client or Siebel Tools, or install only a subset of the components selectable in the installer. Some selections are dependent on other selections.
  - **Production environments.** Install Siebel Enterprise Components behind a firewall. You can also install Siebel Web Client, separately from the server components.

    > **Note:** In a production environment, you install only components meant for user environments. Different security requirements might apply for production versus nonproduction environments. Your security requirements might also vary based on your intended deployment choices (for example, Siebel Application Interface requirements might differ from those for Siebel Server).

- All modules that you install together in a new installation are collocated and use files and directories that exist in the same file structure or are shared with other modules. If you want to keep new installations separate for different modules, then you must install them into different locations.
- You can add components or languages to an existing installation of the same version, as follows:
  - To add components to an existing installation of the same version, you start the Siebel CRM installer, select the existing installation, proceed without selecting any languages, select one or more components to add, and then complete this installation task.
  - To add languages to an existing installation of the same version, you start the Siebel CRM installer, select the existing installation, select any additional languages, and then complete this installation task.
- You can install multiple side-by-side instances of Siebel CRM software on the same computer or operating system instance in separate installation directories. Where multiple instances are installed, when migrating, updating, modifying, or uninstalling an installed instance, you must select the installation directory of the instance to be migrated, updated, modified, or uninstalled.

  > **Note:** As in prior releases, you can configure and use only one Siebel Gateway on each computer or operating system instance, as noted in *General Requirements for Installing and Configuring Siebel Enterprise Components*.

- The `applicationcontainer` directory has been replaced by two directories, as follows:
  - `applicationcontainer_external` (for Siebel Application Interface)
  - `applicationcontainer_internal` (for all other Siebel Enterprise components)

  Correspondingly, multiple Apache Tomcat services (for Windows) or processes (LINUX) are created on the same installation, as needed. These changes are made automatically for updates from prior releases.

- In the Siebel Application Interface installation, Web artifacts for application configurations, which were formerly located in `applicationcontainer\webapps\siebel`, now map to `applicationcontainer_external\siebelwebroot`. The `siebelwebroot` directory contains subdirectories such as `lang` (for each installed language, such as ENU for U.S. English, FRA for French, and so on), `files`, `fonts`, `htmltemplates`, `images`, `migration`, `scripts`, and `smc`.
- You can optionally install new database repository and ancestor database repository support files, as follows:
  - **New Database Repository.** This component must be included if either of the following is true:
    - You need to install new Siebel CRM environments (including development, test, or production)
    - You need to upgrade from a version of Siebel CRM previous to IP2017.
  - **Ancestor Database Repository.** This component must be included for development environments (only) that will be upgraded from a version of Siebel CRM previous to IP2017.
- Oracle Database Client is provided as part of the Siebel CRM product media. You must include the applicable media in the Siebel installation image to be able to install it.
  - Oracle Database Client is included as a required component when you install Siebel Mobile Web Client, to support Oracle Database SE2.
  - You can also install Oracle Database Client manually to use on Siebel Developer Web Client or Siebel Server, to support connections to Oracle Database or LDAP authentication systems, where applicable.

> **Note:** Siebel installs and uses only a single instance of Oracle Database Client on a given operating system instance. The same can be used for all configuration.

Client Installation

Note the following client installation characteristics for Siebel CRM:

- A single installer is now used for both server and client modules. Client modules (Siebel Web Client and Siebel Tools) require the Windows-based installer for installation on Microsoft Windows.
- For a nonproduction environment, you can install Siebel Web Client or Siebel Tools (on Microsoft Windows) separately or collocated with Siebel Enterprise Components. All modules installed together are collocated in the same installation. For a production environment, install Siebel Web Client or Siebel Tools separately from Siebel Enterprise Components.

> **Note:** If you install Siebel Web Client or Siebel Tools with Siebel Enterprise Components in a nonproduction environment, then the files and directories for the client modules are created in the installation directory alongside of those for the server modules, which are listed in *Verifying Installation for Siebel CRM*. If you want to keep new installations separate for different modules, then you must install them into different locations.

- You can install and configure the Siebel Web Client for use as a Mobile Web Client. Oracle Database SE2 and Oracle Database Client are installed automatically for a previous release.

  You must include the Oracle Database SE2 to be able to install and use the Mobile Web Client. Unless updated versions are provided later, it will not be necessary to include the media in subsequent Siebel CRM update installations of later releases.

  > **Note:** If you uninstall Mobile Web Client, then you can also uninstall Oracle Database SE2. Performing a full uninstallation of Siebel Mobile Web Client or restoring a prior release does not uninstall Oracle Database SE2. If you uninstall Siebel Mobile Web Client and also want to uninstall Oracle Database SE2, see the documentation for this product on Oracle Help Center. See also *Siebel Remote and Replication Manager Administration Guide* .

- You can install and configure the Siebel Web Client for use as a Developer Web Client.

Language Installation

Note the following language installation characteristics for Siebel CRM:

- Language installation is more flexible than in previous releases. For Siebel Enterprise Components and Siebel Web Client, you can optionally install other languages, in a new installation or later in an add-language operation. Installing any non-ENU language requires that you have obtained the safeboot ZIP file that includes that language.
- You can add languages to an existing installation of the same version. For more information, see the installation characteristics described earlier in this topic.
- For Siebel Tools, only U.S. English (ENU) is installed and configured, regardless of your language selections that might apply to other installed components.

Configuration

Note the following configuration characteristics for Siebel CRM installation and deployment tasks:

- You can choose whether to provide initial configuration input during Siebel CRM installation, such as to specify application container ports for server modules. If you defer such configuration steps, you can run the Siebel CRM installer again and do them later for the installed software. These tasks must be done within the installer before you can proceed with using Siebel Management Console to complete configuration and deployment tasks.
- As in previous releases, you can choose which installed components to configure and deploy in your topology. It is not necessary to configure and use every component in every installation.

  You might require deploying multiple instances of certain components (such as Siebel Gateway, Siebel Server, or Siebel Application Interface) to support clustering or load balancing. You must keep track of all the components you configure and deploy for your deployment, and their configured settings.

  > **Note:** Use a worksheet, such as what is provided in *Siebel Deployment Planning Worksheet*, to capture installation and configuration settings.

- The TLS port is no longer specified within the installer when you install the Siebel Gateway in a new installation. The Siebel Management Console task for configuring the Gateway profile now includes the TLS Listener Port field.

Uninstallation

Full uninstallation now uses the uninstaller.bat batch file (Windows) or uninstaller script (LINUX), which uninstalls the Siebel CRM software and removes all of the Apache Tomcat services or processes. You can optionally uninstall the current release and restore the installation and configuration of a prior release. For more information, see *Uninstalling Siebel CRM*.

Limitations

The following are some of the applicable limitations for Siebel CRM installation and deployment:

- By definition, migration or update installations are only from lower-versioned software releases. For example, you can update Siebel CRM 21.1 or Siebel CRM 22.1 to any Siebel CRM 22.x release.
- For each Siebel CRM deployment, all installations must be at the same release level (version number).
- Although each server installation now includes all Siebel Enterprise Components, the Siebel CRM installer does not currently support merging existing separate installations of the same or prior versions into a single installation.

Related Topics

Installing Siebel CRM Using the User Interface

About Database Updates for Siebel CRM

Configuring the Siebel Deployment

Planning the Siebel Application Interface Topology

Related Books

Siebel CRM Update Guide and Release Notes on My Oracle Support

Before You Install Siebel CRM

Before you install Siebel CRM, you must perform the preinstallation tasks. Some of these tasks are described in *Siebel Deployment Planning Guide* .

1. Review all documented hardware and software requirements.

   For example, see also *General Guidelines for Installing Siebel CRM*.
2. Review the worksheet in *Siebel Deployment Planning Worksheet* or create your own worksheet, and fill it in as you prepare for installing Siebel CRM.
3. Plan and prepare for your deployment.

4. Configure your RDBMS.

   > **Note:** You can configure the RDBMS at any point before you run the Siebel Database Configuration Wizard. Most of the database tasks described in this guide do not apply if you have an existing Siebel database, such as in an upgrade environment.

   This task applies only for *Roadmap: Installing Siebel CRM for a New Deployment*. For more information, see *Planning RDBMS Installation and Configuration* and *Configuring the RDBMS*.
5. Obtain the Siebel media for Siebel CRM and create a Siebel installation image on the network, from which you will perform installations.

   For more information, see *Creating a Siebel Installation Image*.
6. Create directories for Siebel CRM software and directories for the Siebel File System.

   This step applies only for *Roadmap: Installing Siebel CRM for a New Deployment*. For more information, see *File and Directory Naming Conventions* and *Creating the Siebel File System*.
7. Create the required Siebel accounts.

   This step applies only for *Roadmap: Installing Siebel CRM for a New Deployment*. For more information, see *Creating the Siebel Service Owner Account*.
8. Review all security requirements and take the necessary preparatory steps for your installation case.
9. Perform any manual migration tasks that might be required prior to performing migration installations for an existing deployment, as described in *Additional Tasks for Migration Installations*.

   This step applies only for *Roadmap: Installing Siebel CRM in a Migration Installation Case (Existing Installation of Siebel CRM 16.x or Earlier)*.

General Guidelines for Installing Siebel CRM

Follow these general guidelines for your installations:

- Before installing any Siebel CRM module, review all documented requirements, including those found in *Installation Tasks for Siebel CRM* and in *Requirements for Installing and Configuring Siebel CRM*. Also review the information in the *Siebel Deployment Planning Guide* , the *Siebel Security Guide* , and other relevant guides on *Oracle Help Center*.
- Platform, database, and other certifications for Siebel are found on My Oracle Support. For help searching for certifications, see *How to Check the Certification Matrix in the New My Oracle Support Site* (KB871388).
- You must plan for the installation and management of the Siebel CRM modules that you're installing. Review the installation instructions and the installer screens to help you understand the information requested by the installers. Such information includes installation locations, languages, application container port numbers, other port numbers, and authentication settings.
- When you install Siebel CRM, as described in *Installing Siebel CRM Using the User Interface*, the installer prompts you to specify valid keystore and truststore files, which are Java files containing certificates. These files are necessary for the application container to use secure two-way communications when connecting with other Siebel CRM modules, as occurs during Siebel Management Console configuration and in normal operation. These files must contain the server certificate chain and an imported CA certificate. Generate your files so that the keystore file references both the private key and the public key, while the truststore file references the public key only. Specify the password that was previously configured to open the certificate files. If you don't configure

these files correctly, then you won't be able to configure Siebel CRM, as described in *Configuring Siebel CRM Server Modules*. See also *Siebel Security Guide* .

- Siebel Gateway is installed with every installation of Siebel Enterprise Components. You can configure it to run in the same installation as a Siebel Server or configure it in a separate installation, depending on your requirements. You can optionally install and configure it on multiple nodes in order to support Siebel Gateway clustering. For more information, see *Installation Tasks for Siebel CRM*. See also *Configuring the Siebel Gateway Cluster*.
- You install New Database Repository once, together with the first Siebel Server installed.
- You install all of the Siebel CRM server modules that you'll require to get started, and then you configure them.

  Various requirements apply to the sequence in which you must perform configuration tasks. Security requirements such as for authentication affect many phases of your installation and configuration process, including the overall task sequence. For example, depending on your authentication method, you might need to install the Siebel database for a new deployment before you configure the Siebel CRM server modules using the Siebel Management Console.
- The update installation, migration installation, and database upgrade cases differ in their requirements for installation sequence, Siebel database tasks, configuration of the Siebel environment, and other post-installation tasks.

  For example, for migration installations, you must perform the migration installation of the Siebel Application Interface first (for your existing installation of SWSE). Then perform the migration installation of the Siebel Gateway and Siebel Server.
- You can install additional instances of Siebel Server, Siebel Gateway, or Siebel Application Interface, as needed, for medium-sized or larger deployments.
- You must activate license keys for Siebel CRM, using the License Key Activation utility, after you install a new database, run Incremental Repository Merge (for migration installations), or complete a full database upgrade.
- During a migration installation from a release prior to Siebel CRM 17.0 or an update installation from Siebel CRM 17.0 through Siebel CRM 21.1, the original Siebel CRM installation directory is backed up.

  This directory is renamed from `OriginalName` to `OriginalName_pre22.x`. For example, if the original installation directory name was `ses`, then this directory is renamed to `ses_pre22.7.0.0.0`, indicating a release before Siebel CRM 22.7. The new installation directory is named `OriginalName` , such as `ses`. Retain the original installation directory, in case you choose to restore your prior installation later. For more information, see *Restoring a Prior Installation and Configuration*.

  > **Note:** The installer stops all the associated services for the existing installation so that it can rename the original installation directory. If the installer can't rename the directory for some reason, then it displays an error message. You're prompted to review the possible reasons and resolve the problem before asking the installer to retry the step (renaming the directory). For example, your environment might have running processes (such as Java processes that can be reviewed through the task manager list) related to system environment variables such as JAVA_HOME or might have current file references within the existing installation directory. These processes or references must be terminated or resolved before the home directory renaming can complete.

Related Topics

Installation Tasks for Siebel CRM

Installing Siebel CRM Using the User Interface

Requirements for Installing and Configuring Siebel CRM

About Siebel CRM Releases

Related Books

Siebel Deployment Planning Guide

Siebel Security Guide

Deploying Siebel Open UI

Siebel Performance Tuning Guide

Siebel System Administration Guide

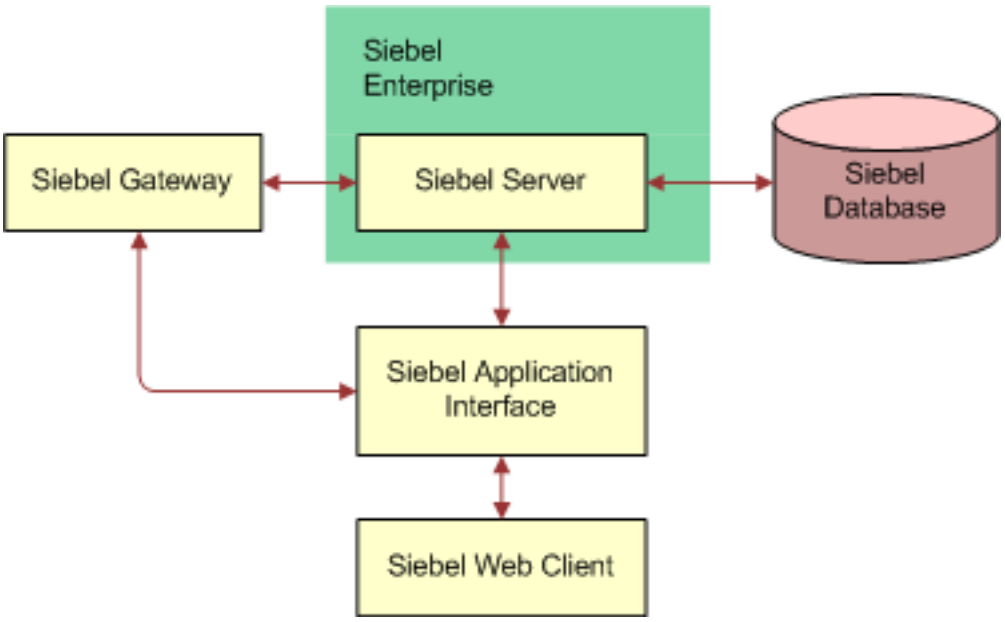
Install & Configure Advisor: Siebel CRM Installer 21.2 & Later, 2753807.2 (Article ID) on My Oracle Support

Siebel CRM Update Guide and Release Notes on My Oracle Support, *KB825124* on My Oracle Support

Overview of Siebel CRM Server Architecture

The following figure illustrates a simplified architecture for the server elements in your Siebel deployment, after you have initially installed and configured the software.

For a more detailed illustration, see *Siebel Deployment Planning Guide* .



The Siebel CRM installation and configuration process requires multiple tasks that you perform in a general sequence that depends on your installation case. For more information, see:

- *Roadmap: Installing Siebel CRM for a New Deployment*
- *Roadmap: Installing Siebel CRM in an Update Installation Case (Existing Installation of Siebel CRM 25.x or Later)*

- *Roadmap: Installing Siebel CRM in a Migration Installation Case (Existing Installation of Siebel CRM 16.x or Earlier)*
- *Roadmap: Installing Siebel CRM in an Upgrade Case (Existing Database Requiring Full Upgrade)*

Platform, database, and other certifications for Siebel are found on My Oracle Support. For help searching for certifications, see *How to Check the Certification Matrix in the New My Oracle Support Site* (KB871388).

Related Topics

Before You Install Siebel CRM

Requirements for Installing and Configuring Siebel CRM

Related Books

Siebel Deployment Planning Guide

Siebel Security Guide

Siebel System Administration Guide

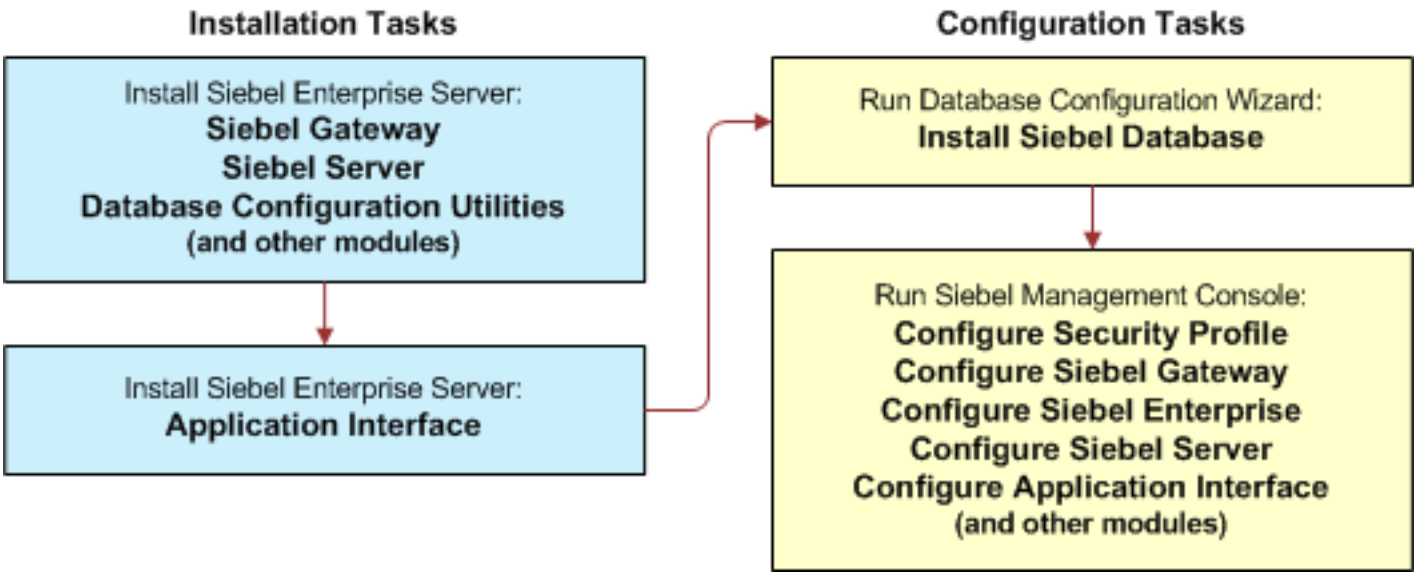
Roadmap: Installing Siebel CRM for a New Deployment

Use this roadmap to guide you as you install and configure Siebel CRM, where there's no existing Siebel CRM installation and no existing Siebel database.

The following figure presents a simplified view of the installation and configuration tasks for Siebel CRM server modules for a new deployment, which are as follows:

1. **Installation Tasks.** Install Siebel Enterprise Server and the following components: Siebel Gateway, Siebel Server, Database Configuration Utilities (and other modules), Siebel Application Interface.
2. **Configuration Tasks.** Run the Database configuration Wizard and install Siebel Database. Run Siebel Management Console and configure the following: Security Profile, Siebel Gateway, Siebel Enterprise, Siebel Server, Siebel Application Interface (and other modules).

Note: This illustration doesn't mention installation or configuration tasks for security or the Siebel Migration application. However, those tasks, where applicable, are implicitly part of this roadmap.



Unless you use another authentication method, a Siebel database must exist before you can configure the Siebel CRM modules, in order to provide an authentication framework for the configuration tasks. (Database Configuration Utilities is no longer a selectable option for new installations. Instead, you install New Database Repository with the first Siebel Server.)

Note: You install Siebel CRM before you install the Siebel database or configure the Siebel CRM products. For more information, see *Installation-Related Changes in Siebel CRM*.

The installation process for Siebel CRM for a new deployment requires multiple tasks that you perform in the following general sequence:

1. Before performing installation and configuration tasks, review all documented requirements.

   See *Before You Install Siebel CRM*.
2. Create the network image for the current Siebel CRM release.

   See *Creating a Siebel Installation Image*.
3. Install Siebel CRM. You can select the following Siebel CRM modules to install:

   - Siebel Enterprise Components (includes Siebel Server, Siebel Gateway, Siebel Application Interface, and other components)
   - New Database Repository (requires installing Siebel Enterprise Components)
   - Ancestor Database Repository (requires installing Siebel Enterprise Components)
   - Siebel Web Client
   - Siebel Tools

   See *Siebel CRM Download and Installation*. In particular, see *Installing Siebel CRM in a New Installation*.

4. Choose the following configuration options, depending on your installation selections:
    - Application Interface Container Configuration
    - Enterprise Container Configuration
    - Siebel Web Client Configuration (Mobile Web Client or Developer Web Client)

    For these configuration options, you specify authentication information and port numbers that will be used by your Siebel CRM deployment. The port numbers must be unique on each installation computer or operating system instance. You must perform these tasks before you do Siebel Management Console configuration tasks.

    See *Siebel CRM Download and Installation*.
5. (Optional) Install Siebel Enterprise Components on additional nodes, such as if you will use Siebel Gateway clustering.

    See *Configuring the Siebel Gateway Cluster*.
6. Verify the installation of Siebel CRM modules.

    See *Verifying and Troubleshooting Your Installation*.
7. Configure the RDBMS.

    See *Configuring the RDBMS*.
8. Install the Siebel database.

    This task installs a Siebel database for Siebel CRM. This database contains schema changes, Siebel Repositories, and seed data for Siebel CRM functionality through the current release.

    See *Installing the Siebel Database on the RDBMS*.
9. Activate the license keys.

    See *Activating License Keys*.
10. Run the Siebel Management Console and configure your Siebel deployment:
    a. Configure a security profile.
    b. Configure the Siebel Gateway. (Configure the primary Siebel Gateway, for a cluster environment.)
    c. (Optional) Configure the Siebel Gateway cluster.
    d. Configure the Siebel Enterprise.
    e. Configure the Siebel Server.
    f. Configure the Siebel Application Interface.
    g. Perform any other configuration tasks that you require for your deployment.

    See *Configuring Siebel CRM Server Modules*.
11. Perform additional postinstallation tasks.

    See *Additional Postinstallation and Configuration Tasks*.
12. (Optional) Install additional Siebel CRM modules. For example:
    - Desktop Integration Siebel Agent (DISA)
    - Siebel Search software
    - Siebel Test Execution
    - Siebel Approval Manager

13. (Optional) Install and configure additional instances of Siebel Gateway, Siebel Server, or Siebel Application Interface if they are required for your deployment, then verify each installation.
14. Perform any remaining postinstallation tasks. For example, install any required third-party products.

    See *Additional Postinstallation and Configuration Tasks*.

Platform, database, and other certifications for Siebel are found on My Oracle Support. For help searching for certifications, see *How to Check the Certification Matrix in the New My Oracle Support Site* (KB871388).

Related Topics

Installation-Related Changes in Siebel CRM

Roadmap: Installing Siebel CRM in an Update Installation Case (Existing Installation of Siebel CRM 25.x or Later)

Roadmap: Installing Siebel CRM in a Migration Installation Case (Existing Installation of Siebel CRM 16.x or Earlier)

Roadmap: Installing Siebel CRM in an Upgrade Case (Existing Database Requiring Full Upgrade)

Related Books

Siebel CRM Update Guide and Release Notes on My Oracle Support

Roadmap: Installing Siebel CRM in an Update Installation Case (Existing Installation of Siebel CRM 17.x or Later)

Use this roadmap to guide you in performing the installation tasks for installing the latest Siebel CRM release where there's an existing installation of Siebel CRM 25.x or later. This is the update installation case.

The installation process for Siebel CRM for an update installation case requires multiple tasks that you perform in the following general sequence:

1. Before performing installation and configuration tasks, review all documented requirements.

   See *Before You Install Siebel CRM*.
2. (Strongly recommended) Back up the Siebel database before installing Siebel CRM.

   > **Note:** If, after installing Siebel CRM, allowing the PostInstallDBSetup utility to run, and optionally running RepositoryUpdate, you decide to restore a prior release, as described in *Restoring a Prior Installation and Configuration*, then, as part of this task, you can use this backup to restore the Siebel database to its prior state.

3. Create the network image for the current Siebel CRM 22.x release.
4. For each existing Siebel Enterprise Server installation, install Siebel CRM as an update installation.

   See *Siebel CRM Download and Installation*. In particular, see *Installing Siebel CRM in an Update Installation*.

   > **Note:** The installer also runs the PostInstallDBSetup utility. You can also run the optional RepositoryUpdate utility to support new features that require repository updates but do not use nonextensible objects. Some other postinstallation tasks for the Siebel database might also apply. For more information, see *About Database Updates for Siebel CRM*.

5. Verify the installation of Siebel CRM modules.

   See *Verifying and Troubleshooting Your Installation*.
6. For each existing Siebel Tools or Siebel Web Client installation, install Siebel CRM as an update installation.

   > **Note:** If it is not already present, then Oracle Database SE2 is installed for the local database, for Siebel Mobile Web Client.

   See *Siebel CRM Download and Installation*.
7. Verify the installation of Siebel Tools and Siebel Web Clients.

   See *Verifying and Troubleshooting Your Installation*.

Platform, database, and other certifications for Siebel are found on My Oracle Support. For help searching for certifications, see *How to Check the Certification Matrix in the New My Oracle Support Site* (KB871388).

Related Topics

Installation-Related Changes in Siebel CRM

Roadmap: Installing Siebel CRM for a New Deployment

Roadmap: Installing Siebel CRM in a Migration Installation Case (Existing Installation of Siebel CRM 16.x or Earlier)

Roadmap: Installing Siebel CRM in an Upgrade Case (Existing Database Requiring Full Upgrade)

About Database Updates for Siebel CRM

Related Books

Siebel CRM Update Guide and Release Notes on My Oracle Support

Roadmap: Installing Siebel CRM in a Migration Installation Case (Existing Installation of Siebel CRM 16.x or Earlier)

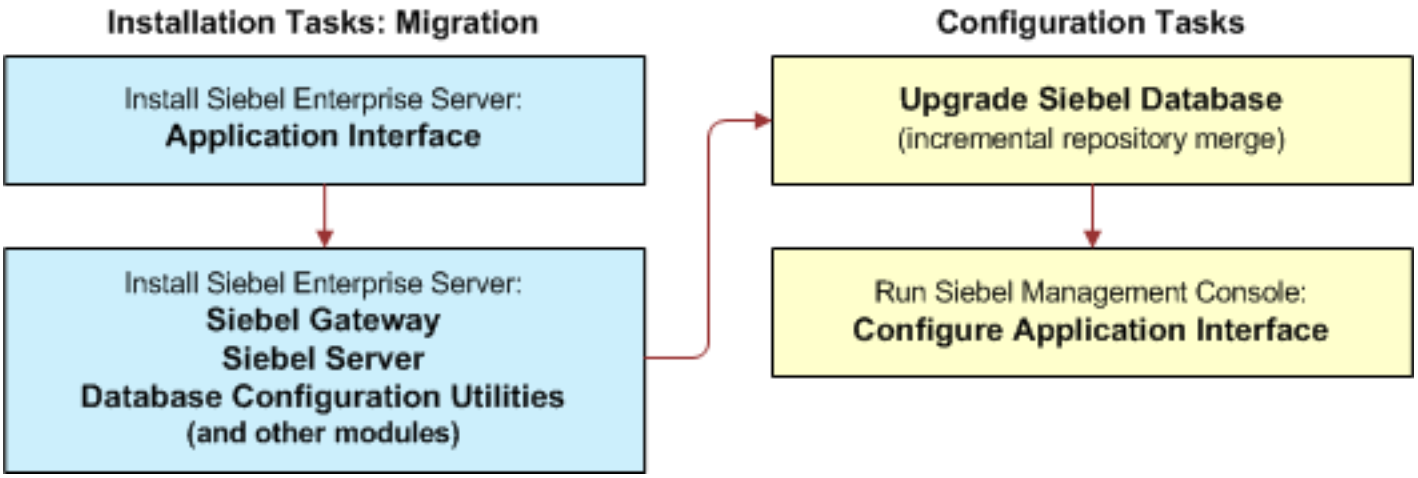
Use this roadmap to guide you in performing the installation tasks for deploying Siebel CRM where there is an existing installation of a prior release of Siebel CRM 16.x or earlier (prior to Siebel CRM 17.0). You can migrate from any release as early as Siebel CRM 8.1.1 or 8.2.2. This is the migration installation case.

> **Tip:** For the database upgrade case, where you are upgrading from a Siebel CRM release prior to version 8.1.1 or version 8.2.2, see *Roadmap: Installing Siebel CRM in an Update Installation Case (Existing Installation of Siebel CRM 25.x or Later)* instead. In an upgrade case, you perform a full database upgrade instead of running Incremental Repository Merge (IRM). In general, this book uses the term *migration* to refer to a migration installation, with IRM, and uses the term *upgrade* to refer to a full database upgrade.

The following figure presents a simplified view of the installation and configuration tasks for Siebel CRM server modules for a migration deployment, which are as follows:

1. **Installation Tasks: Migration.** Install Siebel Enterprise Server and the following components: Siebel Application Interface, Siebel Gateway, Siebel Server, Database Configuration Utilities (and other modules).
2. **Configuration Tasks.** Upgrade Siebel Database (incremental repository merge). Run Siebel Management Console and Siebel Application Interface.

Note: This illustration does not mention installation or configuration tasks for security or the Siebel Migration application. However, those tasks, where applicable, are implicitly part of this roadmap.



Note: For each applicable installed Siebel CRM module of a qualified version, you install Siebel CRM as a migration installation. Complete the installation before you run Incremental Repository Merge or configure the Siebel CRM products. For more information, see *Installation-Related Changes in Siebel CRM*.

This roadmap only summarizes the migration installation process and does not include details about required tasks. For example, you must do the following tasks after performing the migration installation tasks. See also *Additional Tasks for Migration Installations*.

- **Upgrade the database client.** You might need to upgrade the database client software to a supported version for Siebel CRM for nonOracle database clients. Starting with Siebel CRM 25.2, Oracle database clients are automatically updated as part of Siebel install. Siebel installer will install the Oracle database client on all platforms by default under a new folder named "Oracledbclient" under `<SIEBEL_HOME>/oracledbclient` folder. See *About Configuring Siebel CRM*.
- **Run Incremental Repository Merge (IRM).** After you perform the migration installation for a Siebel Server where you perform database tasks and for Siebel Tools, you must run Incremental Repository Merge. For more information, see *Siebel Database Upgrade Guide* .
- **Reset passwords in Siebel Gateway and on the Siebel Server.** Customers must reset any passwords on the Siebel Gateway that were previously encrypted using encryption other than AES. Such passwords are now encrypted using AES. For more information about reencrypting these passwords, see *Siebel Security Guide* . See also *Additional Tasks for Migration Installations*.

The installation process for Siebel CRM in a migration installation case requires multiple tasks that you perform in the following general sequence:

1. Before performing installation tasks, review all documented requirements and perform any necessary premigration tasks.

See *Before You Install Siebel CRM* and *Additional Tasks for Migration Installations*.

2. Create the network image for the current Siebel CRM 22.x release.
3. Remove the existing configuration on the Web server computer where you previously installed Siebel Web Server Extension (SWSE).

   > **CAUTION:** You must perform this task before you perform the migration installation of Siebel Application Interface for the existing installation of SWSE.

   See *Additional Tasks for Migration Installations* and *About Configuring Siebel CRM*.
4. For each existing installation of SWSE, install Siebel CRM as a migration installation. Siebel Application Interface is installed in its place.

   As part of installation, also specify authentication information and port numbers that will be used by your Siebel CRM deployment. The port numbers must be unique on each installation computer or operating system instance.

   See *Siebel CRM Download and Installation*. In particular, see *Installing Siebel CRM in a Migration Installation*.
5. For all other existing installations of Siebel Enterprise Server modules, install Siebel CRM as a migration installation. Also perform any necessary postmigration tasks.

   As part of installation, also specify authentication information and port numbers that will be used by your Siebel CRM deployment. The port numbers must be unique on each installation computer or operating system instance.

   > **Note:** When you migrate the Siebel Gateway, you also must provide information to allow your existing configurations to migrate. This includes the port number of the existing Siebel Gateway Name Server as well as some Siebel Application Interface settings that you specified in Step 4. (If you installed multiple instances of Siebel Application Interface, then specify the settings that apply to the instance of Siebel Application Interface from which you will run Siebel Management Console.)

   See *Siebel CRM Download and Installation*. See also *Additional Tasks for Migration Installations*.
6. (Optional) Install Siebel Gateway on additional nodes, if you will use clustering.

   See *Configuring the Siebel Gateway Cluster*.
7. Verify the installation of Siebel CRM modules.

   See *Verifying and Troubleshooting Your Installation*.
8. For each existing installation of Siebel Tools, install Siebel CRM as a migration installation. Also perform any necessary postmigration tasks.

   See *Siebel CRM Download and Installation*. See also *Additional Tasks for Migration Installations*.
9. For each existing installation of Siebel Web Client, install Siebel CRM as a migration installation. Also perform any necessary post migration tasks.

   > **Note:** Oracle Database SE2 is installed for use with the local database (for Mobile Web Client). Oracle Database Client is automatically installed as part of Siebel.

   See *Siebel CRM Download and Installation*. See also *Additional Tasks for Migration Installations*.
10. Verify the installation of Siebel Tools and Siebel Web Clients.

    See *Verifying and Troubleshooting Your Installation*.

11. Run Incremental Repository Merge on the Siebel database.

    See *Siebel Database Upgrade Guide* .
12. Run the Siebel Management Console and configure your Siebel deployment:

    a. (Optional) Configure the Siebel Gateway cluster.
    b. Configure the Siebel Application Interface.
    c. Perform any other configuration tasks that you require for your deployment.

    See *Configuring Siebel CRM Server Modules*.
13. Perform additional postinstallation tasks for the Siebel Server.

    See *Additional Postinstallation and Configuration Tasks*.
14. (Optional) Install additional Siebel CRM modules. For example:

    - Desktop Integration Siebel Agent (DISA)
    - Install Siebel Search software
    - Siebel Test Execution
    - Siebel Approval Manager
15. (Optional) Install and configure additional instances of Siebel Gateway, Siebel Server, or Siebel Application Interface if they are required for your deployment, then verify each installation.
16. Perform any remaining postinstallation tasks. For example, install or upgrade any required third-party products.

    See *Additional Postinstallation and Configuration Tasks*.

Platform, database, and other certifications for Siebel are found on My Oracle Support. For help searching for certifications, see *How to Check the Certification Matrix in the New My Oracle Support Site* (KB871388).

Related Topics

Installation-Related Changes in Siebel CRM

Roadmap: Installing Siebel CRM for a New Deployment

Roadmap: Installing Siebel CRM in an Update Installation Case (Existing Installation of Siebel CRM 25.x or Later)

Roadmap: Installing Siebel CRM in an Upgrade Case (Existing Database Requiring Full Upgrade)

Related Books

Siebel Database Upgrade Guide

Siebel CRM Update Guide and Release Notes on My Oracle Support

Roadmap: Installing Siebel CRM in an Upgrade Installation Case (Existing Database Requiring Full Upgrade)

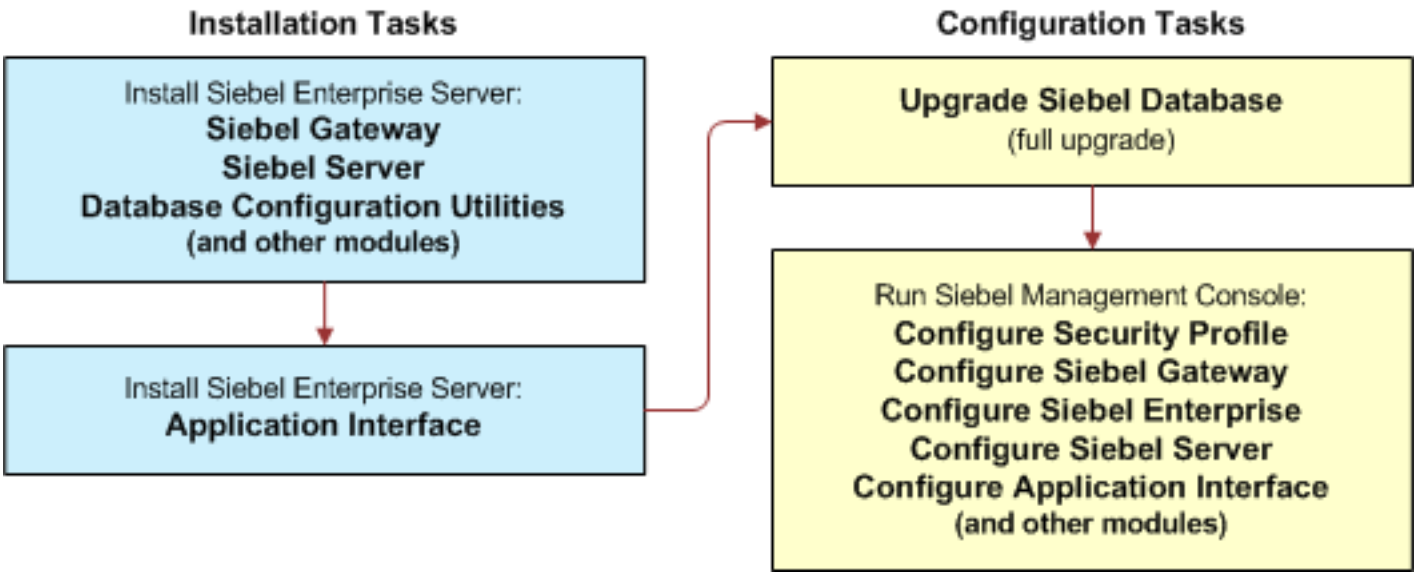
Use this roadmap to guide you in performing the installation and configuration tasks for deploying Siebel CRM where there's an existing Siebel database and you're upgrading from a Siebel CRM release before Siebel CRM 8.1.1 or 8.2.2. Where a full database upgrade must be performed, you can perform either new or migration installations (where supported), before you perform the database upgrade tasks. This roadmap assumes that you're performing new installations.

> **Tip:** For the migration installation case, where you're upgrading from an existing installation of a prior release of Siebel CRM, see *Roadmap: Installing Siebel CRM in a Migration Installation Case (Existing Installation of Siebel CRM 16.x or Earlier)* instead. However, for this roadmap, you perform a full database upgrade instead of running Incremental Repository Merge (IRM). In general, this book uses the term *migration* to refer to a migration installation (usually implying IRM) and uses the term *upgrade* to refer to a full database upgrade.

The following figure presents a simplified view of the installation and configuration tasks for Siebel CRM server modules for an upgrade case, which are as follows: (Database Configuration Utilities is no longer a selectable option for new installations. Instead, you install New Database Repository with the first Siebel Server. For upgrades, also install Ancestor Database Repository.)

1. **Installation Tasks.** Install Siebel Enterprise Server and the following components: Siebel Gateway, Siebel Server, Database Configuration Utilities (and other modules), Siebel Application Interface.
2. **Configuration Tasks.** Upgrade Siebel Database (full upgrade). Run Siebel Management Console and configure the following: Security Profile, Siebel Gateway, Siebel Enterprise, Siebel Server, Siebel Application Interface (and other modules).

> **Note:** This illustration doesn't mention installation or configuration tasks for security or the Siebel Migration application. However, those tasks, where applicable, are implicitly part of this roadmap.



Note: For each applicable Siebel CRM module, you install Siebel CRM as a new installation, before you upgrade the Siebel database or configure the Siebel CRM products. For more information, see *Installation-Related Changes in Siebel CRM*.

This roadmap only briefly mentions the Siebel database upgrade tasks and places them in the context of the other steps. Before you upgrade the database, you must have installed at least one Siebel Server with New Database Repository and Ancestor Database Repository and also installed Siebel Tools. For more information about upgrading, see *About Installing in Upgrade Environments* and see *Siebel Database Upgrade Guide* .

The installation process for Siebel CRM in an upgrade installation case requires multiple tasks that you perform in the following general sequence:

1. Before performing installation and configuration tasks, review all documented requirements.

   See *Before You Install Siebel CRM*.
2. Create the network image for the current Siebel CRM 22.x release.

   See *Creating a Siebel Installation Image*.
3. Install Siebel CRM and perform other applicable steps.

   For details, see *Roadmap: Installing Siebel CRM for a New Deployment* and *Siebel CRM Download and Installation*.
4. After verifying the installation, upgrade the Siebel database.

   See *About Installing in Upgrade Environments* and see *Siebel Database Upgrade Guide* .
5. Perform additional applicable steps in *Roadmap: Installing Siebel CRM for a New Deployment*.

Platform, database, and other certifications for Siebel are found on My Oracle Support. For help searching for certifications, see *How to Check the Certification Matrix in the New My Oracle Support Site* (KB871388).

Related Topics

Installation-Related Changes in Siebel CRM

Roadmap: Installing Siebel CRM for a New Deployment

Roadmap: Installing Siebel CRM in an Update Installation Case (Existing Installation of Siebel CRM 25.x or Later)

Roadmap: Installing Siebel CRM in a Migration Installation Case (Existing Installation of Siebel CRM 16.x or Earlier)

About Installing in Upgrade Environments

Related Books

Siebel Database Upgrade Guide

Siebel CRM Update Guide and Release Notes on My Oracle Support

3 Siebel CRM Download and Installation

Siebel CRM Download and Installation

This chapter explains how to install Siebel CRM. It includes the following topics:

- *Finding and Downloading a Siebel CRM Release/Update Image Content*
- *Installation Tasks for Siebel CRM*
- *Installing Siebel CRM Using the User Interface*
- *Installing Siebel CRM Using Silent Mode*
- *Additional Tasks for Migration Installations*

Finding and Downloading a Siebel CRM Release/Update Image Content

This topic describes how to find and download a Siebel CRM release or update.

Patches and Updates tab on My Oracle Support

You can locate a Siebel CRM patch on My Oracle Support:

1. Go to the Patches & Updates Tab
2. Go to Patch Search
3. On the left side choose Product or Family
    - **Product** – Choose Siebel CRM
    - **Release** – Choose the desired release or update from the LOV
    - **Platform –** Choose your desired platform"(optional) -. For example: Microsoft Windows.
4. Click on the Search button.
5. Click on the desired Description of your product
6. Click on the **Readme** button to review the readme for the desired download and review the Required Downloads section of the file for your platform. The Archive files section will outline the content of the zip file downloads that are available.
7. Choose and download desired zip files using the instructions mentioned in *Creating a Siebel Installation Image*.

Note: Available releases and updates are made available on the site above.

Oracle Software Delivery Cloud

Oracle software Delivery cloud is not currently used for Siebel releases.

Downloading patches for specific support related issues.

Downloads related to service requests are provided by technical support and generally follow the same pattern as downloads for releases and updates.

About the Siebel Installation Image

For each Siebel CRM release or update, you must download the Siebel CRM media files and create the Siebel image from which you install Siebel CRM software.

Content will be extracted from the download ZIP files to yield the Siebel CRM product content that is specific to your Siebel CRM release and specific to your supported operating system (Windows, Linux, or IBM AIX). The specific ZIP files and their contents vary for each operating system. (All customers require the Windows-based media in order to install Siebel client modules, though Windows and UNIX-based images must remain separate.)

Media packaging has been optimized and reorganized to reflect the changes stemming from the new installer. Media content is optimized to support the new installer and include modules not previously part of Siebel CRM media or installation. For example:

- For all installation cases, only media for the current release of Siebel CRM are required.
- Media are consolidated for server and client modules.
- Media are provided, for installing non-ENU safeboot files.
- Media for new database repository and ancestor database repository support files are provided for inclusion in Siebel CRM server installations, where required for your deployment.
- Media for Oracle Database SE2 are provided for inclusion in Siebel Web Client installations (for Siebel Mobile Web Client).
- Media for Oracle Database Client are provided as part of the Siebel binaries and do not need to be separately downloaded for both clients and server.
- Media for Siebel Mobile SDK and Siebel Smart Answer are now provided. These modules do not use the new Siebel CRM installer and are not documented in this guide.

Creating a Siebel Installation Image

This topic describes how to create the Siebel image from which installations will be performed.

Use a standard unpacking tool to extract from each download ZIP file the corresponding Siebel CRM product content. Validate that the unpacking tool that you are using can handle files of the sizes of the Siebel media files.

Identifying the needed content for a Siebel installation image

The following table identifies the Siebel CRM product content contained within the download ZIP files. This product content makes up the Siebel image from which you run the Siebel CRM installer for each Siebel CRM release. You can skip downloading any item that you do not need for the specific release you are installing. The items shown in the table are applicable to the current Siebel CRM release. For the specific mapping of download ZIP files and Siebel CRM product content, consult the readme file.

Siebel CRM Product Content	Requirement
README.html and other documents.	These documents identify the ZIP file mapping and provide additional information to help you install.
Siebel CRM binary files (specific to operating system).	Yes. Required for every Siebel CRM release you install.
Non-ENU safeboot files (specific to operating system).	Required where any languages other than U.S. English (ENU) are to be installed and deployed.
The current Siebel Repository. Applies to all supported operating systems and RDBMS platforms. Corresponds to the New Database Repository selection in the installer.	Required for new database installations and for database upgrades from releases prior to Siebel CRM 17.0.
The existing ancestor database repository: previous Siebel Repository versions from which you can upgrade. Applies to all supported operating systems and RDBMS platforms. Corresponds to the Ancestor Database Repository selection in the installer. For information about supported database upgrade paths, see *Siebel Database Upgrade Guide* .	Required for database upgrades from releases prior to Siebel CRM 17.0.
Files for Oracle Database SE2, used for the local database and Siebel Mobile Web Client. (Do not extract the Oracle_Database_ SE2.zip file after you create the Siebel image. The installer does that during Siebel Web Client installation.)	Required for Siebel Web Client installations. Not required subsequently where already installed, unless an updated version is provided.
Installer for Siebel Smart Answer. This installer is separate from the Siebel CRM installer and does not have to reside in the same image location. For more information about installing and using this module, see *Siebel Smart Answer Guide* .	Required for installing Siebel Smart Answer.

To create a Siebel installation image

1. Download the ZIP files that you require for the current release of Siebel CRM. Download these ZIP files into a single empty directory in which you will store these ZIP files. For example, for Siebel CRM 25.2, you might create a directory named like one of the following:

```
(Windows) C:\Siebel_media_25.x\Siebel_ZIP_25.2
(UNIX) /Siebel_media_25.x/Siebel_ZIP_25.2
```

> **Note:** *Determining the Locations for the Siebel Media Files and the Siebel Image* describes a scheme for creating directories to contain the ZIP files and the Siebel installation image files.

2. For each operating system and Siebel CRM release, create a designated directory to serve as the Siebel image location on that platform. For example, for Siebel CRM 25.2, you might create a directory named like one of the following:

   ```
   (Windows) C:\Siebel_media_25.x\Siebel_image_25.2
   (LINUX) /Siebel_media_25.x/Siebel_image_25.2
   ```

   > **Note:** *Determining the Locations for the Siebel Media Files and the Siebel Image* describes a scheme for creating directories to contain the ZIP files and the Siebel installation image files.

3. Copy all applicable ZIP files into the image directory that you created in Step 2: those that are specific to that operating system or that apply to all operating systems.

   > **Note:** Do not combine files for different operating systems in the same directory.

4. Extract the contents of the ZIP files in this directory. Use “extract here” or the equivalent to yield the Siebel CRM image.

   > **Note:** If you are prompted about overwriting existing files or directories when extracting content from multiple ZIP files after the first one, then you must allow the overwrites.

   The Siebel image is created in the directory you created in Step 2. You will run the Siebel CRM installer from the `Disk1\Install` subdirectory.

You must allocate disk space on the network for the Siebel images that you will use as the source location for Siebel CRM installations. Retain all of the Siebel images until they no longer apply. Also retain all of the Siebel media files (ZIP files).

The directory in which you create a Siebel image must meet the following requirements:

- The user creating the Siebel image must have write permission in the Siebel image directory. Users who have to access the Siebel image directory in order to perform installations must have read and execute permissions. Users who have to modify files to support unattended installation mode must also have write permission. See also the relevant topics in *Planning Your Siebel Deployment*.
- Any directory that you specify for creating a Siebel image must contain sufficient disk space for all of the Siebel CRM modules that you include in the Siebel image.
- Any directory that you specify for creating a Siebel image must follow the conventions described in *File and Directory Naming Conventions*. For example, on Windows, a directory name must not contain spaces or number signs (pound or hash signs).
- (Windows) Any directory that you specify for creating a Siebel image must have a drive letter assigned to it.
- Do not move or rename any subdirectories within the Siebel image, such as the `Disk1`subdirectory. It is acceptable to move or rename the top-level directory in which you created the Siebel image.

> **Note:** After you create the Siebel image in a central location, you can optionally copy the entire version-specific Siebel image to a computer where you will install the software. Doing so can improve performance during the installation, particularly for installations of Siebel Enterprise Components. For additional recommendations or requirements for installers, see Oracle Universal Installer documentation on Oracle Technology Network. See also *Requirements for Installing and Configuring Siebel CRM* .

Related Topics

Installing and Using Oracle Database SE2 for the Local Database

Installing and Deploying Siebel CRM with Multiple Languages

About Siebel CRM Releases

General Requirements for Installing and Configuring Siebel Enterprise Components

Determining the Locations for the Siebel Media Files and the Siebel Image

It is recommended that you download or copy all of the Siebel media files to a central location on your network where you have sufficient disk space. This might or might not be the same location where you create your Siebel image. For example, for Siebel CRM 22.x releases, you might create the top-level directory `siebel_media_22.x` to store all of the Siebel CRM files. Then create additional directories, like the following examples for Siebel CRM 22.7:

- Create a subdirectory `siebel_ZIP_22.7`, in which to store the Siebel CRM download ZIP files. Do not combine files for different operating systems or releases in the same directory.
- Create another subdirectory, `siebel_image_22.7`, a version-specific Siebel CRM directory that will be the location of the Siebel image.

In this example, you copy the ZIP files into the Siebel image directory and then extract the contents, which represent the Siebel image for the installable Siebel CRM modules. For more information about creating the Siebel installation image, see *Creating a Siebel Installation Image*.

Your directories for the ZIP files and the Siebel image might be as shown in the following table. For other Siebel CRM releases, substitute the appropriate version numbers or create similar directories.

Directory Function	Example Directory Locations	Comments
ZIP files directory	`(Windows) C:\Siebel_media_22.x\Siebel_ZIP_22.7` `(UNIX) /Siebel_media_22.x/Siebel_ZIP_22.7`	Create a version-specific subdirectory like this example to contain the ZIP files that you obtain from Oracle for Siebel CRM.
Siebel image directory	`(Windows) C:\Siebel_media_22.x\Siebel_image_22.7` `(UNIX) /Siebel_media_22.x/Siebel_image_22.7`	Create a version-specific subdirectory like this example to contain the Siebel image for each release. You run the Siebel CRM installer from this location.

Note: After you create the Siebel image in a central location, you can optionally copy the entire version-specific Siebel image to a computer where you will install the software. Doing so can improve performance during the installation, particularly for installations of Siebel Enterprise Components. For additional recommendations or requirements for installers, see Oracle Universal Installer documentation on Oracle Technology Network. See also *Requirements for Installing and Configuring Siebel CRM*.

Cross-Platform Issues When Creating Siebel Image Directories

Each Siebel image that you create contains directories representing the specified operating system platforms that you'll use, such as Windows, AIX, and so on. The Siebel image itself can reside on any supported operating system.

The operating systems that Siebel CRM media are provided for, and on which you can install Siebel CRM, are:

- Microsoft Windows
- IBM AIX
- Linux

Many customers support multiple operating systems. For example, a customer who installs server modules on UNIX still might need to install client or server modules that run only on a supported Microsoft Windows operating system, such as Siebel Web Client and Siebel Tools. If you support multiple operating systems (such as Windows and one of the supported UNIX operating systems), then use one or more of the following strategies when creating your Siebel images:

- **Create a single, multiplatform Siebel image on one of your operating systems.** For example, you might create Siebel images for Windows as well as AIX, where all of the files reside on Windows. Use a cross-platform networking tool, such as Samba, to make the Siebel image accessible from operating systems other than the one where the Siebel image was created. In order to install modules on AIX computers (in this example), users who perform installations must be able to access the Siebel image files in the Windows location. This approach is generally recommended, because it consolidates all of the modules and applicable operating systems in a single Siebel image.
- **On each applicable operating system, create a separate Siebel image that includes the modules that will be installed on that operating system.** In order to install the Siebel CRM modules (in this example), no cross-platform tool would be necessary to access the Siebel image files, because the files for each module already reside on the operating system on which the installations will be performed.

Installer Errors and the Siebel Image

You must validate that the Siebel image was created correctly for all of the applicable modules. However, installation errors sometimes indicate problems in the Siebel image.

If, when you run a Siebel installer, errors are returned about missing or corrupt files, then remove all of the Siebel image files you extracted from the ZIP files, make sure that you have all of the ZIP files that you require, and then extract the contents again to recreate the Siebel image.

For installation instructions, see *Siebel CRM Download and Installation*.

For installation requirements, see *Requirements for Installing and Configuring Siebel CRM*.

For installation troubleshooting information, see *Verifying and Troubleshooting Your Installation*.

About Installation Tasks for Siebel CRM

Use the Siebel CRM installer to download and install Siebel CRM server and client modules, including Siebel Gateway, Siebel Server, Siebel Application Interface, Siebel Web Client, Siebel Tools, and other modules. This chapter describes new installations, update installations, and migration installations.

Before installing Siebel CRM modules, review the following and perform any necessary preinstallation tasks:

- *Before You Install Siebel CRM* and other topics in *Overview of Installing Siebel CRM*
- *Additional Tasks for Migration Installations*
- *About Database Updates for Siebel CRM*
- *Requirements for Installing and Configuring Siebel CRM*

After a new installation, you install the Siebel database and then you configure Siebel CRM modules using the Siebel Management Console. After a migration installation, you run Incremental Repository Merge (IRM) on the Siebel database.

Use the instructions in this chapter to install the following Siebel CRM modules:

- Siebel Enterprise Components (includes Siebel Application Interface, Siebel Server, Siebel Gateway, and other modules)
- New Database Repository (requires installing Siebel Enterprise Components)
- Ancestor Database Repository (requires installing Siebel Enterprise Components)
- Siebel Web Client
- Siebel Tools

You can install and deploy multiple instances of Siebel CRM modules. For example:

- You can optionally install Siebel Gateway on multiple nodes in order to support Siebel Gateway clustering. For more information, see *Configuring the Siebel Gateway Cluster*.
- You can optionally install Siebel Server on multiple nodes in order to support Siebel Server load balancing. For more information, see *About Installing Additional Siebel Servers for an Existing Siebel Enterprise*.
- You can optionally install Siebel Application Interface on multiple nodes in order to configure load balancing for these Siebel Application Interface instances. For more information, see *Planning the Siebel Application Interface Topology*.

You perform migration installations using the same installer that you use for new installations. A migration installation replaces your existing installation with a new installation. The installation location, installation languages, server configuration data, and other settings from the existing installation are used for the new installation, where feasible. Before the new installation is performed, the existing installation is moved to a backup location and can no longer be used.

> **CAUTION:** Before you install Siebel Application Interface as a migration installation, you must remove the existing configuration for the Siebel Web Server Extension (SWSE) from Siebel CRM 16.x or earlier. If you are using data encryption, then you must back up the key file. For more information, see *Additional Tasks for Migration Installations*.

Note: After completing all of the migration installations, you must configure the Siebel Application Interface. A migration installation of Siebel Application Interface does not copy customer configurations or files from your existing installation of SWSE to the new installation.

Related Topics

Installing Siebel CRM Using the User Interface

Installing Siebel CRM Using Silent Mode

Additional Tasks for Migration Installations

Configuring Siebel CRM Server Modules

Requirements for Installing and Configuring Siebel CRM

About Installing Siebel CRM

Installing Siebel CRM

This topic provides instructions for installing Siebel CRM modules for the current release, using the GUI installation method, as part of standard deployment. Use these instructions to install the following modules:

- Siebel Enterprise Components (includes Siebel Server, Siebel Gateway, Siebel Application Interface, and other modules)
- New Database Repository
- Ancestor Database Repository
- Siebel Web Client
- Siebel Tools

For more information about installing these modules, see *Installation Tasks for Siebel CRM*. See also *Requirements for Installing and Configuring Siebel CRM*.

Note: This topic includes instructions for installing Siebel CRM modules in a new installation, update installation, or migration installation. For an existing installation of Siebel CRM 16.x or earlier, you perform a migration installation instead of a new installation. For an existing installation of Siebel CRM 17.0 through Siebel CRM 22.1, you perform an update installation.

Use the installation procedure for your installation case to run the Siebel CRM installer to install Siebel CRM. Which options and screens appear in your installation session depends on the following:

- Your installation case. For migration or update installations, the screens that appear also depend on the specific version of your existing Siebel CRM deployment.
- The prior selections you have made.

- The contents of the installation image you created.

> **Note:** If you know you do not need a given optional module, then you do not need to include it in the installation image. Optional modules that are part of the installation image but were previously installed and are not required for subsequent releases are not reinstalled unless an updated version is provided. If your installation image does not include these, then the corresponding or dependent installation options will not be present in the installer. For example, you cannot install Siebel Web Client unless the Siebel image includes the media for Oracle Database SE2.

This topic includes the following information:

- *Pre-Installation Tasks*
- *Installing Siebel CRM in a New Installation*
- *Installing Siebel CRM in an Update Installation*
- *Installing Siebel CRM in a Migration Installation*
- *Verifying Installation*

Related Topics

Installation Tasks for Siebel CRM

Installing Siebel CRM Using Silent Mode

Configuring Siebel CRM Server Modules

Additional Postinstallation and Configuration Tasks

Verifying Installation for Siebel CRM

Uninstalling Siebel CRM

Requirements for Installing and Configuring Siebel CRM

Related Books

Siebel Security Guide

Siebel Deployment Planning Guide

Siebel Global Deployment Guide

Preinstallation Tasks

Note the following information before you begin installing Siebel CRM. This topic is part of *Installing Siebel CRM Using the User Interface*.

- Download the product media for Siebel CRM 22.x and create the Siebel network image for the release. Run the installers from this image. Include the ZIP files for the modules you will require for your installations, and make the necessary selections in the installer when you perform the installations.
- Siebel Gateway and Siebel Server services are automatically stopped when you install or uninstall Siebel CRM.

- Auto-Start Services are available for the Siebel Gateway Server and Siebel Server. To enable this functionality, set the AUTOSTART_SERVICES parameter to Yes or No in:
  - `Disk1\install\oneclick.properties` (to auto-start services after installing Siebel CRM)
  - `<PROD_HOME>\oui\bin\patchrollback.properties` (to auto-start services after restoring a prior release)

  Note: The AUTOSTART_SERVICES parameter setting also applies if you want to auto-start services after restoring a prior release.

- Review the database-related tasks that might apply after you install Siebel CRM. For more information, see *Installing the Siebel Database on the RDBMS*. See also *Siebel Database Upgrade Guide* .

Installing Siebel CRM in a New Installation

Use the following procedure to install Siebel CRM in a new installation. This task is part of *Installing Siebel CRM Using the User Interface*.

Note: Use this procedure only where Siebel CRM is not already installed. Where a prior release of Siebel CRM 17.x or later is already installed, you perform an update installation instead, as described in *Installing Siebel CRM in an Update Installation*. Where Siebel CRM 16.x or earlier is already installed, you perform a migration installation instead, as described in *Installing Siebel CRM in a Migration Installation*.

This task is a step in:

- *Roadmap: Installing Siebel CRM for a New Deployment*
- *Roadmap: Installing Siebel CRM in an Upgrade Case (Existing Database Requiring Full Upgrade)*

To install Siebel CRM modules in a new installation

1. Review *Before You Install Siebel CRM* and *Installation Tasks for Siebel CRM* and observe the requirements and recommendations.

   For example, if you are installing Siebel Enterprise components that will communicate with an existing configured installation of Siebel Gateway on another computer or operating system instance, then make sure that the Siebel Gateway is running.

   You can stop any Siebel Server or Siebel Gateway services running on the same computer where you are installing Siebel CRM. If you do not stop them, the installer will stop them for you.
2. Log on to the server computer:
   - (Windows) Log on to the server computer, using an account with administrative privileges, such as the Siebel service owner account that you recorded in the copy that you made earlier of the worksheet in *Siebel Deployment Planning Worksheet*. The account that you use must belong to the Windows domain of your Siebel CRM deployment and must have full write permissions to the Siebel File System.
   - (LINUX) Log on to the server computer, using the Siebel service owner account that you recorded in the copy that you made earlier of the worksheet in *Siebel Deployment Planning Worksheet*.
3. (Windows) Shut down all of the open programs that are running on the server.

4. Open a new window or shell. Navigate to the Siebel image location for your Siebel CRM 22.x release. For more information, see *Creating a Siebel Installation Image*.
   - (Windows) In Windows Explorer, navigate to the directory where the installer is located. Navigate to `Siebel_Image\Disk1\install`. In this path, `Siebel_Image` is the directory for your version-specific Siebel network image, such as `C:\Siebel_Install_Image\25.2`.
   - (LINUX) In the shell you opened, navigate to the directory where the installer is located. For example, for Linux, navigate to `Siebel_Image/Disk1/install`. In this path, Siebel_Image is the directory for your version-specific Siebel network image, such as `/Siebel_Install_Image/25.2`.
5. Verify the permissions to the directory into which you will install Siebel CRM. You must have write and execute permission.
6. (Windows) Before you run the Siebel CRM installer to perform the migration installation of a Siebel CRM module from a release prior to Siebel CRM 17.0, run the following command:

   ```
   module.exe -debug -silent -attachhome ORACLE_HOME_NAME=Oracle_Home_Name ORACLE_HOME=Oracle_Home
   ```

   For example, you might enter a command like this, depending on the Oracle home name for the module you are installing and on the original installation directory:

   ```
   module.exe -debug -silent -attachhome ORACLE_HOME_NAME=SES_HOME ORACLE_HOME=c:\Siebel\ses
   ```

7. Start the Siebel CRM installer for the current release:
   - (Windows) To start the installer, double-click setup.bat.
   - (UNIX) To start the installer, enter the following command:

     ```
     ./runInstaller.sh -invPtrLoc inventory_location
     ```

   In this command, inventory_location is the full path of the oraInst.loc file (for example,`/export/home/oraInst.loc`), which points to the OUI inventory location. Use the -invPtrLoc option if you want to specify an inventory pointer file other than the default.

   > **Note:** Optionally, for a silent (unattended) installation, you can append the flags described in the information about installing Siebel CRM in silent mode that is described in *Installing Siebel CRM Using Silent Mode*.

   The Siebel Installation Location screen appears.
8. Specify the directory in which to install the software, and click Next.
   - (Windows) You might specify an installation path for a new installation like `C:\Siebel`, `C:\Siebel_SES`, `C:\Siebel_AI`, `C:\Siebel_Client`, `C:\Siebel_Tools`, or a similar value.
   - (LINUX) You might specify an installation path for a new installation like `/Siebel`, `/Siebel_SES`, `/Siebel_AI`, `C:\Siebel_Client`, `C:\Siebel_Tools`, or a similar value. Siebel Web Client and Siebel Tools can be installed only on Microsoft Windows.

     Accept the default path or specify a different fully qualified path by typing it directly or by clicking Browse. If you have an existing installation, then you must specify a new path for the new installation. For more information about installation path requirements, see *File and Directory Naming Conventions*.

9. For 25.10 and above a popup requests confirmation whether this is a new installation for a hardware migration scenario or not.

   If it is part of hardware migration scenario, in which the older version of Siebel would have been installed on other machine which is getting replaced by a new hardware. User also needs to confirm the existing encryption level if it is part of hardware migration scenario.

   If you are adding languages or components to an existing installation of the same version, select the directory of the existing installation for which you want to add languages or components.

   The Language Selection screen appears.
10. Select the languages that you want to install, and click Next.

    All of the supported languages are available for selection for a new installation. You must select at least one language in order to proceed. To select a language, you click the name of the language in the Available box and then use the arrow controls to move it into the Selected box. Use Ctrl+click to select multiple languages.

    > **Note:** You specify the languages to install in order to run Siebel applications using these languages. The files that are specific to the languages chosen in this step will be copied to your installation. Language installation characteristics vary for different modules. For more information about language installation, see also *Installation-Related Changes in Siebel CRM*.

    If you are adding languages to an existing installation of the same version, select the languages you want to add and proceed to complete the installation.

    > **Note:** When you configure the Siebel Server or the Siebel Application Interface in Siebel Management Console, you can specify which installed languages to deploy and which installed language is the primary language. Deploy the superset of all of the languages required by the Application Object Manager components you require in your Siebel deployment.

    If you are adding components to an existing installation of the same version, proceed to the next step without selecting any languages.

    The Component Selection screen appears.
11. For an installation into a non-development environment, such as production or test environment, do the following. For more information, see *Installation-Related Changes in Siebel CRM*. For a development environment, skip to Step 11.

    a. Specify which components to install. Select at least one of the following:

       - **Siebel Enterprise Components.** Includes Siebel Gateway, Siebel Server, Siebel Application Interface, and EAI Connector.
       - **Siebel Web Client.** You can install and configure Siebel Web Client for use as either a Mobile Web Client or a Developer Web Client.
       - **New Database Repository.** For non-development environments, this is only required for new database installations (not Upgrades nor Updates to existing Siebel databases). When required,

it need only be installed on one of the Siebel Servers in the Enterprise, but may optionally be installed on others.

For Siebel Mobile Web Client, Oracle Database SE2 must have been included when you prepared the product media in *Creating a Siebel Installation Image*, and are installed with Siebel Web Client. It may be automatically updated with later Siebel updates.

If you are adding components to an existing installation of the same version, select the components to add and proceed to complete the installation.

b. Specify which configuration tasks to perform later in the installation session. Select one or more of the following (available selections are based on the modules you are installing):

- **Enterprise Container Configuration.** See Step 12 through Step 14.
- **Application Interface Container Configuration.** See Step 12 through Step 14.
- **Siebel Web Client Configuration.** See Step 15.

c. Go to Step 12.

12. For an installation into a development environment, do the following. For more information, see *Installation-Related Changes in Siebel CRM*.

a. Specify which components to install. Select one or more of the following:

- **Siebel Enterprise Components.** Includes Siebel Gateway, Siebel Server, Siebel Application Interface, and EAI Connector.
- **New Database Repository.** For development environments, this is required for new database installations and Upgrades from pre-IP2017 versions. When required, it need only be installed on one of the Siebel Servers in the Enterprise, but may optionally be installed on others.
- **Ancestor Database Repository.** This is required in development environments for all customers who are Upgrading Siebel from a version of Siebel CRM previous to Innovation Pack 2017. It is only required on the Siebel Server or Siebel Tools machine from which the Upgrade will be run.
- **Siebel Web Client.** You can install and configure Siebel Web Client for use as either a Mobile Web Client or a Developer Web Client.

  For Siebel Mobile Web Client, Oracle Database SE2 must have been included when you prepared the product media in *Creating a Siebel Installation Image*, and is installed with Siebel Web Client. Oracle Database client is automatically installed and updated as part of Siebel install. It will not be installed again for subsequent updates you install.
- **Siebel Tools.** Installs the Siebel Tools client.

If you are adding components to an existing installation of the same version, select the components to add and proceed to complete the installation.

b. Specify which configuration tasks to perform later in the installation session. Select one or more of the following (available selections are based on the modules you are installing):

- **Enterprise Container Configuration.** See Step 12 through Step 14.
- **Application Interface Container Configuration.** See Step 12 through Step 14.
- **Siebel Web Client Configuration.** See Step 15.
- **Siebel Tools Configuration.** See Step 16.

c. Click Next.

If you selected Enterprise Container Configuration or Application Interface Container Configuration, then the Application Container Ports screen appears.

13. Specify port numbers and click Next.

    Siebel Application Interface Port Details:

    - **HTTPS Redirect Port.** (Default: 443)
    - **HTTP Connection Port.** (Default: 80)
    - **Shutdown Port.** (Default: 8005)

    Siebel Enterprise Components Port Details:

    - **HTTPS Redirect Port.** Used for secure HTTPS connections to the application container, which can include connections redirected from the HTTP port. (Default: 444)
    - **HTTP Connection Port.** Incoming connections on the HTTP port are redirected to the secure HTTPS port. (Default: 8081)
    - **Shutdown Port.** Used for management purposes by Siebel CRM, for local use only. (Default: 8006)

    Note: Specifying these port numbers allows the application containers and other parts of the configuration framework to work together during and after the configuration and deployment tasks you perform using Siebel Management Console. The port numbers must be unique on each installation computer or operating system instance. See also *Configuring Siebel CRM Server Modules*.

    If you selected Application Interface Container Configuration in a previous step, then the Application Interface Authentication screen appears.

14. Specify the appropriate authentication credentials for access to Siebel Management Console, and click Next.

    - **User Name.** Specify the user name to use when you log in to Siebel Management Console, as described in *Starting the Siebel Management Console*.
    - **Password.** Specify the password for this user name.
    - **Confirm Password.**
    - **Application Context Name.** Specify the name of the application context as defined in Siebel Application Interface. This name appears in all of the URLs for this installation. The default value is siebel. You can also change this value for an existing installation. For more information, see *Customizing URLs for Siebel CRM Applications*.

    Note: You use the credentials specified here when you first log in to Siebel Management Console, after installing the first Siebel Application Interface node. These credentials, which are stored in the applicationinterface.properties file, are valid only before you have configured the security profile. After configuring security, you must log in to Siebel Management Console again with valid credentials for the authentication method specified in the security profile. Then, when you install any subsequent instance of Siebel Application Interface, you would also specify those same valid credentials (that are used by the first Siebel Application Interface node) for the specified authentication method, for example, user name *sadmin* and password *mypwd*.

    If you selected Enterprise Container Configuration or Application Interface Container Configuration in a previous step, then the Client Authentication screen appears.

15. Specify information that supports secure communications for the installed software, and click Next. The installer does not require that separate files are specified. Use files that meet your security requirements.

    Siebel Enterprise Components Authentication:

    - **Keystore Name.** Specify a file (such as a JKS file) you have generated that will serve as the keystore.
    - **Truststore Name.** Specify a file (such as a JKS file) you have generated that will serve as the truststore.
    - **Password.** Specify the password for the specified keystore and truststore files.
    - **Confirm Password.**

    Siebel Application Interface Authentication:

    - **Keystore Name.**
    - **Truststore Name.**
    - **Password.**
    - **Confirm Password.**

    > **Note:** The keystore and truststore files are Java files containing certificates. They are necessary for the application container to use secure two-way communications when connecting with other Siebel CRM modules, as occurs during Siebel Management Console configuration and in normal operation. These files must contain the server certificate chain and an imported CA certificate. Generate your files so that the keystore file references both the private key and the public key, while the truststore file references the public key only. Specify the password that was previously configured to open the certificate files. Use the same password for the keystore and truststore files. Use fully qualified domain names rather than IP addresses. If you do not configure these files correctly, then you will not be able to configure Siebel CRM, as described in *General Guidelines for Installing Siebel CRM*. See also *Siebel Security Guide* .

    (Windows) If you are installing and configuring Siebel Web Client, then the Siebel Web Client Configuration details screen appears.

16. (Windows) Specify the Siebel Web Client Configuration details, and click Next.

    - **Client Type.** Select either Mobile Web Client or Developer Web Client.
    - **(Mobile Web Client only) Siebel Remote Server Hostname.** Specify the name of the Siebel Server on which you are running the Siebel Remote server components. The Siebel Remote server components are required for initializing and synchronizing the local database on the Siebel Mobile Web Client computer.
    - **Select Required Applications.** Specify the Siebel applications your Siebel Web Client users require. Click to select one application. Ctrl+click to select additional applications.
    - **Developer Web Client options:**
        - **Database Server.** Select the type of database server that you are using for Siebel CRM. Choose one of the following:
            - Oracle Database Server
            - IBM DB2 UDB for Windows and UNIX
            - Microsoft SQL Server (select this for Azure SQL Database Servers)
            - IBM DB2 UDB for z/OS
        - **Database details.** Specify database information according to your database type. Enter the data that you previously specified in the worksheet in *Siebel Deployment Planning Worksheet*.
            - For Oracle Database, specify the Database Alias and Table Owner.
            - For IBM DB2, specify the Database Alias, Table Owner, and DB2 Instance Home Directory.
            - For Microsoft SQL Server, specify the Database Server Host Name and Database Instance Name.
            - For IBM DB2 for z/OS, specify the Database Alias, Schema Qualifier (Table Owner), Current SQLID, and DB2 Instance Home Directory.
            - For Microsoft SQL Server and Azure SQL Database Server.
        - **Directory Path for Siebel File System.** Specify the directory path for a network-based Siebel File System.

          > **Note:** For the Siebel File System path, you can use a UNC share name (for example, `\\SRV1\siebfile`) or a mapped drive (for example, `K:\siebelFS`). For more information, see *File and Directory Naming Conventions*. See also *Creating the Siebel File System*.

        - **Gateway Server Address.** Specify the network name or the IP address of the computer on which the Siebel Gateway is installed. To enter a specific port number, append the Gateway Server Address string with a colon and the desired port number.
        - **Enterprise Server Name.** Specify the name of the Siebel Enterprise to which this Siebel Developer Web Client will connect for administration purposes. The Enterprise Server name is the name of the Siebel Enterprise under which the Siebel Servers that support the server database operate.
        - **Request Server Name.** Specify the name of the Siebel Server on which the Server Request Broker component is operating. This component is used for dynamic assignment and other interactive operations.
        - **Search Server Name.** Specify the host name of the server computer where the search server operates.
        - **Search Server Port Number.** Specify the port number of the search server. The default value is 2048.

- **HTTP Connection Port.** Incoming connections on the HTTP port are redirected to the secure HTTPS port. (Default: 9001)
- **Shutdown Port.** Used for management purposes, for local use only. (Default: 9005)

(Windows) If you are installing and configuring Siebel Tools, then the Siebel Tools Configuration details screen appears.

17. (Windows) Specify the Siebel Tools Configuration details, and click Next.

    - **Database Server.** Select the type of database server that you are using for Siebel CRM. Choose one of the following:

      - Oracle Database Server
      - IBM DB2 UDB for Windows and UNIX
      - Microsoft SQL Server (select this for Azure SQL Database Servers)
      - IBM DB2 UDB for z/OS

      > **Note:** Depending on your selection, the Siebel CRM installer validates that the correct database connectivity software has been installed. If it has not been installed, then you must exit the installer, install the required software, and then restart the installer.

    - **Database details.** Enter the following information for your database server. Enter the same data that you put in your *Siebel Deployment Planning Worksheet*.

      - For Oracle Database, specify the Database Alias and Table Owner.
      - For IBM DB2, specify the Database Alias and Table Owner.
      - For Microsoft SQL Server or Azure SQL Database Server, specify the Database Server Host Name and Database Instance Name.
      - For IBM DB2 for z/OS, specify the Database Alias (as defined in DB2 Connect) and Table Owner or Schema Qualifier.

    - **Directory path for Siebel File System.** Specify the directory path for a network-based Siebel File System by typing it directly or by clicking Browse.

      > **Note:** You can use a UNC share name (for example, \\SRV1\siebfile) or a mapped drive (for example, K:\siebelFS). For more information, see *File and Directory Naming Conventions* and *Creating the Siebel File System*.

    The Summary screen appears.

18. In the Summary screen, review the information presented.

    This screen displays the location where Siebel CRM will be installed. Before you proceed, confirm that you have met all of the installation requirements, such as for disk space. Then do one of the following:

    - To begin installing Siebel CRM, click Install. You can also click Install after saving a response file.
    - To save a response file to use for a silent installation later, click Save Response File, and then save the file from the dialog box that appears. If you are not also installing at this time, then click Cancel after you save the response file. For information about performing silent installations, see *Installing Siebel CRM Using Silent Mode*.

    The Installation Progress screen appears. Siebel CRM is installed into the directory that you specified in Step 8.

19. When the Next button becomes available, the installation is complete. Click Next.

    The Finish Installation screen appears. This screen indicates whether the installation was successful and shows the location of the installer log file. You can access this file to review the installation results.
20. To close the installer, click Close.

    This new installation of Siebel CRM is now finished.

    If you installed New Database Repository for a new installation of Siebel Enterprise Components, then the shortcut for the Database Configuration Wizard is created in the program folder, using a generated name.

    As required for your installation, you must configure your deployment using the Siebel Management Console, as described in *Configuring Siebel CRM Server Modules*. (If you did not complete Enterprise Container Configuration or Application Interface Container Configuration, then you must run the Siebel CRM installer again and do so before you run Siebel Management Console.)

Installing Siebel CRM in an Update Installation

Use the following procedure to install Siebel CRM in an update installation. This task is part of *Installing Siebel CRM Using the User Interface*.

> **Note:** Use this procedure only where a prior release of Siebel CRM 17.x or later is already installed. For a new installation, see *Installing Siebel CRM in a New Installation*. Where Siebel CRM 16.x or earlier is already installed, you perform a migration installation instead, as described in *Installing Siebel CRM in a Migration Installation*.

> **CAUTION:** If you are updating an installed release of Siebel CRM 17.x or 18.x (through 18.4) to the current release, then you must update all installed instances of Siebel Application Interface before you update other installed Siebel CRM modules. If you are updating Siebel CRM 18.5 or later to the current release, then you can update the installed modules in any sequence.

This task is a step in *Roadmap: Installing Siebel CRM in an Update Installation Case (Existing Installation of Siebel CRM 25.x or Later)*.

To install Siebel CRM modules in an update installation

1. Review *Before You Install Siebel CRM* and *Installation Tasks for Siebel CRM* and observe the requirements and recommendations.

   For example, if you are installing Siebel Enterprise components that will communicate with an existing configured installation of Siebel Gateway on another computer or operating system instance, then make sure that the Siebel Gateway is running.

   You can stop any Siebel Server or Siebel Gateway services running on the same computer where you are installing Siebel CRM. If you do not stop them, the installer will stop them for you.

2. (Strongly recommended) Back up the Siebel database before installing Siebel CRM.

   Note: After installing Siebel CRM, allowing the PostInstallDBSetup utility to run, and optionally running RepositoryUpdate, if you decide to restore your prior release, as described in *Restoring a Prior Installation and Configuration*, then, as part of this task, you can use this backup to restore the Siebel database to its prior state.

3. (Strongly recommended) Back up the Siebel Gateway registry before installing Siebel CRM.

   Note: After installing Siebel CRM, if you decide to restore your prior release, as described in *Restoring a Prior Installation and Configuration*, then, as part of this task, you can use this backup to restore the Siebel Gateway registry to its prior state.

4. Log on to the server computer:
   - (Windows) Log on to the server computer, using an account with administrative privileges, such as the Siebel service owner account that you recorded in the copy that you made earlier of the worksheet in *Siebel Deployment Planning Worksheet*. The account that you use must belong to the Windows domain of your Siebel CRM deployment and must have full write permissions to the Siebel File System.
   - (UNIX) Log on to the server computer, using the Siebel service owner account that you recorded in the copy that you made earlier of the worksheet in *Siebel Deployment Planning Worksheet*.
5. (Windows) Shut down all of the open programs that are running on the server.
6. Open a new window or shell. Navigate to the Siebel image location for your Siebel CRM 22.x release. For more information, see *Creating a Siebel Installation Image*.
   - (Windows) In Windows Explorer, navigate to the directory where the installer is located. Navigate to `Siebel_Image\Disk1\install`. In this path, `Siebel_Image` is the directory for your version-specific Siebel network image, such as `C:\Siebel_Install_Image\22.7`.
   - (LINUX) In the shell you opened, navigate to the directory where the installer is located. For example, for Linux, navigate to `Siebel_Image/Disk1/install`. In this path, Siebel_Image is the directory for your version-specific Siebel network image, such as `/Siebel_Install_Image/22.7`.
7. Start the Siebel CRM installer for the current release:
   - (Windows) To start the installer, double-click setup.bat.
   - (UNIX) To start the installer, enter the following commands:

     ```
     dbenv.sh./runInstaller.sh -invPtrLoc inventory_location
     ```

     In this command, inventory_location is the full path of the oraInst.loc file (for example,`/export/home/oraInst.loc`), which points to the OUI inventory location. Use the -invPtrLoc option if you want to specify an inventory pointer file other than the default.

     Note: Optionally, for a silent (unattended) installation, you can append the flags described in the information about installing Siebel CRM in silent mode that is described in *Installing Siebel CRM Using Silent Mode*.

   The Siebel Installation Location screen appears.
8. Select the location of your existing installation from the dropdown list, and click Next.
   If you are performing an update installation from a prior release of Siebel CRM that supports the new Siebel CRM installer (such as Siebel CRM 21.2, the first release to support the new installer), then the Backup Location screen appears.
9. Specify the Backup Location details, and click Next.

The default backup directory location is `SIEBEL_ROOT\mde_backup`. You can specify another location for this directory. A version-specific backup directory is created under this location when you perform an update installation from the prior release that supports the new Siebel CRM installer. For example:

- If you are updating a release that does not support the new installer (Siebel CRM 17.0 through Siebel CRM 21.1) to a release that does support the new installer (such as Siebel CRM 22.7), then the backup directory is created as `SIEBEL_ROOT_pre22.7.0.0.0`, such as `SES_pre22.7.0.0.0`. This directory, which contains the complete installation, is created at the same level as the existing `SIEBEL_ROOT` directory. For uninstallation cases that involve restoring Siebel CRM 21.1 or earlier, see *Uninstalling Siebel CRM*.

  > **Note:** For a prior installation of Siebel Web Client (from Siebel CRM 20.8 through Siebel CRM 21.1), the backup that is created might be fairly large, because it also includes a full copy of Oracle Database SE2.

- If you are updating a release that supports the new installer (for example, updating Siebel CRM 21.4 to Siebel CRM 22.7), then the backup directory might be named `SIEBEL_ROOT_21.4`, such as `SIEBEL_CRM_21.4` (for an installation directory of `SIEBEL_CRM`). This directory contains custom files and customer-modified files such as configuration files, which you might require if you must restore your prior release. For uninstallation cases that involve restoring Siebel CRM 21.2 or later, see *Uninstalling Siebel CRM*.

The Database Parameters screen appears.

10. Specify the database configuration details, and click Next.

    In this screen, you specify input for the PostInstallDBSetup utility, which runs after you perform an update installation from any prior release of Siebel CRM 17.0 or later. For more information, see *About Database Updates for Siebel CRM*.

11. A screen for encryption parameters is seen if the user choose to skip the post install database setup execution in the database configuration screen.

    User must provide the required database details on this new screen to proceed further. If the user choose to defer or execute the post install database setup, the encryption parameter screen gets disabled as the required inputs are already available through database configuration details screen.

    The Summary screen appears.

12. In the Summary screen, review the information presented.

    This screen displays the location where Siebel CRM will be installed. Before you proceed, confirm that you have met all of the installation requirements, such as for disk space. Then do one of the following:

    - To begin installing Siebel CRM, click Install. You can also click Install after saving a response file.
    - To save a response file to use for a silent installation later, click Save Response File, and then save the file from the dialog box that appears. If you are not also installing at this time, then click Cancel after you save the response file. For information about performing silent installations, see *Installing Siebel CRM Using Silent Mode*.

    The Installation Progress screen appears. Siebel CRM is installed into the directory that you specified in Step 8.

13. When the Next button becomes available, the installation is complete. Click Next.

    The Finish Installation screen appears. This screen indicates whether the installation was successful and shows the location of the installer log file. You can access this file to review the installation results.

14. To close the installer, click Close.

    This update installation of Siebel CRM is now finished.

Installing Siebel CRM in a Migration Installation

Use the following procedure to install Siebel CRM in a migration installation. This task is part of *Installing Siebel CRM Using the User Interface*.

> **Note:** Use this procedure only where a prior release of Siebel CRM 16.x or earlier is already installed. For a new installation, see *Installing Siebel CRM in a New Installation*. Where a prior release of Siebel CRM 17.x or later is already installed, you perform an update installation instead, as described in *Installing Siebel CRM in an Update Installation*.

This task is a step in:

- *Roadmap: Installing Siebel CRM in a Migration Installation Case (Existing Installation of Siebel CRM 16.x or Earlier)*
- (Optional) *Roadmap: Installing Siebel CRM in an Upgrade Case (Existing Database Requiring Full Upgrade)*

For more information about migration installations, see *Installation Tasks for Siebel CRM*.

> **Note:** If you need to add any modules that were not present in your original installation, including, for example, new modules or additional instances of Siebel Server or Siebel Application Interface, then you must install them as a new installation, as described in *Installing Siebel CRM in a New Installation*.

> **CAUTION:** You must perform the migration installation for Siebel Application Interface before you perform the migration installations for any other Siebel CRM server modules. When you do the migration installations for other Siebel Enterprise Server modules, the installer prompts you to provide the host name and installation location for the migrated installation of Siebel Application Interface. Without this information, the migration installations of Siebel Gateway, Siebel Server, or other modules cannot complete successfully.

To install Siebel CRM in a migration installation

1. Review *Before You Install Siebel CRM* and *Installation Tasks for Siebel CRM* and observe the requirements and recommendations.
2. Perform any premigration tasks that might be necessary, as noted in *Additional Tasks for Migration Installations*.

   > **CAUTION:** For example, you must remove the existing configuration for the SWSE before you install Siebel Application Interface as a migration installation. If you are using data encryption, then you must back up the original key file.

   If you are installing Siebel Enterprise components that will communicate with an existing configured installation of Siebel Gateway on another computer or operating system instance, then make sure that the Siebel Gateway is running.
3. (Windows) For a migration installation of Siebel Web Client from a release prior to Siebel CRM 16.0, exit Siebel QuickStart for Mobile Web Client.

   > **Note:** Siebel QuickStart is no longer supported as of Siebel CRM 16.0. For a Siebel Mobile Web Client using the Siebel QuickStart feature, you must exit the QuickStart agent if it is running before you perform the migration installation. To do this, right-click the QuickStart icon in the system tray, and then choose Exit.

4. (Optional) Shut down all of the open programs associated with the Siebel CRM modules to be updated on this computer.
   For example, stop any services (for the existing installed instance of a prior version) for the Siebel Gateway Name Server, Siebel Server, Siebel Management Agent, or Oracle Configuration Manager. Also shut down any running instances of the Siebel Server Manager (srvrmgr). If you do not shut down these services, then the installer shuts them down. (Siebel Management Agent and Oracle Configuration Manager are not installed or supported in the current Siebel CRM release.)
5. (Windows) For a migration installation of Siebel Application Interface for the existing installation of SWSE, stop the Microsoft IIS services by doing the following:
   - Stop the IIS Administration service.
   - Stop the World Wide Web Publishing service.
6. (UNIX) For a migration installation of Siebel Application Interface for the existing installation of SWSE, stop the Web server by running one of the following commands:
   - **`ompmnctl stopall`** – for Oracle HTTP Server (on supported Linux operating systems)
   - **`stopapa`** – for other Apache-based Web servers (on supported operating systems)
7. Log on to the server computer:
   - (Windows) Log on to the server computer, using an account with administrative privileges, such as the Siebel service owner account that you recorded in the copy that you made earlier of the worksheet in *Siebel Deployment Planning Worksheet*. The account that you use must belong to the Windows domain of your Siebel CRM deployment and must have full write permissions to the Siebel File System.
     For Siebel Web Client, log onto the client computer as a user with administrative privileges.
   - (UNIX) Log on to the server computer, using the Siebel service owner account that you recorded in the copy that you made earlier of the worksheet in *Siebel Deployment Planning Worksheet*.
8. Open a new window or shell. Navigate to the Siebel image location for your Siebel CRM 22.x release.
   - (Windows) In Windows Explorer, navigate to the directory where the installer is located. Navigate to **`Siebel_Image\Disk1\install`**. In this path, **`Siebel_Image`** is the directory for your version-specific Siebel network image, such as **`c:\Siebel_Install_Image\22.7`**.
   - (UNIX) In the shell you opened, navigate to the directory where the installer is located. For example, for Linux, navigate to **`Siebel_Image/Disk1/install`**. In this path, Siebel_Image is the directory for your version-specific Siebel network image, such as **`/Siebel_Install_Image/22.7`**.
9. (Windows) Before you run the Siebel CRM installer to perform the migration installation of a Siebel CRM module, run the following command:

   ```
   module.exe -debug -silent -attachhome ORACLE_HOME_NAME=Oracle_Home_Name ORACLE_HOME=Oracle_Home
   ```

   For example, you might enter a command like this, depending on the Oracle home name for the module you are installing and on the original installation directory:

   ```
   module.exe -debug -silent -attachhome ORACLE_HOME_NAME=SES_HOME ORACLE_HOME=c:\Siebel\ses
   ```

10. Start the Siebel CRM installer for the current release:
    - (Windows) To start the installer, double-click setup.bat.
    - (LINUX) To start the installer, enter the following command:

      ```
      ./runInstaller.sh -invPtrLoc inventory_location
      ```

In this command, inventory_location is the full path of the oraInst.loc file (for example,`/export/home/oraInst.loc`), which points to the OUI inventory location. Use the -invPtrLoc option if you want to specify an inventory pointer file other than the default.

> **Note:** Optionally, for a silent (unattended) installation, you can append the flags described in the information about installing Siebel CRM in silent mode that is described in *Installing Siebel CRM Using Silent Mode*.

The Siebel Installation Location screen appears.

11. Select the location of your existing installation from the dropdown list, and click Next.

The Application Container Port Configuration screen appears.

12. Specify port numbers for this installation, and click Next.

Siebel Application Interface Port Details:

- **HTTPS Redirect Port.** (Default: 443)
- **HTTP Connection Port.** (Default: 80)
- **Shutdown Port.** (Default: 8005)

Siebel Enterprise Components Port Details:

- **HTTPS Redirect Port.** Used for secure HTTPS connections to the application container, which can include connections redirected from the HTTP port. (Default: 444)
- **HTTP Connection Port.** Incoming connections on the HTTP port are redirected to the secure HTTPS port. (Default: 8081)
- **Shutdown Port.** Used for management purposes by Siebel CRM, for local use only. (Default: 8006)

> **Note:** Specifying these port numbers allows the application containers and other parts of the configuration framework to work together during and after the configuration and deployment tasks you perform using Siebel Management Console. The port numbers must be unique on each installation computer or operating system instance. See also *Configuring Siebel CRM Server Modules*.

The Application Interface Authentication screen appears.

13. Specify the appropriate authentication credentials for access to Siebel Management Console, and click Next.

- **User Name.** Specify the user name to use when you log in to Siebel Management Console, as described in *Starting the Siebel Management Console*.
- **Password.** Specify the password for this user name.
- **Confirm Password.**

> **Note:** You use the credentials specified here when you first log in to Siebel Management Console, after installing the first Siebel Application Interface node. These credentials, which are stored in the applicationinterface.properties file, are valid only before you have configured the security profile. After configuring security, you must log in to Siebel Management Console again with valid credentials for the authentication method specified in the security profile. Then, when you install any subsequent instance of Siebel Application Interface, you would also specify those same valid credentials (that are used by the first Siebel Application Interface node) for the specified authentication method, for example, user name *sadmin* and password *mypwd*.

The Client Authentication screen appears.

14. Specify information that supports secure communications for the installed software, and click Next. The installer does not require that separate files are specified. Use files that meet your security requirements.

    Siebel Enterprise Components Authentication:

    - **Keystore Name.** Specify a file (such as a JKS file) you have generated that will serve as the keystore.
    - **Truststore Name.** Specify a file (such as a JKS file) you have generated that will serve as the truststore.
    - **Password.** Specify the password for the specified keystore and truststore files.
    - **Confirm Password.**

    Siebel Application Interface Authentication:

    - **Keystore Name.**
    - **Truststore Name.**
    - **Password.**
    - **Confirm Password.**

    Note: The keystore and truststore files are Java files containing certificates. They are necessary for the application container to use secure two-way communications when connecting with other Siebel CRM modules, as occurs during Siebel Management Console configuration and in normal operation. These files must contain the server certificate chain and an imported CA certificate. Generate your files so that the keystore file references both the private key and the public key, while the truststore file references the public key only. Specify the password that was previously configured to open the certificate files. Use the same password for the keystore and truststore files. Use fully qualified domain names rather than IP addresses. If you do not configure these files correctly, then you will not be able to configure Siebel CRM, as described in *General Guidelines for Installing Siebel CRM*. See also *Siebel Security Guide* .

    The screens in Step 15 through Step 18 contain settings that are required in order to migrate your existing configurations and allow you to use Siebel Management Console for any additional configuration tasks.

    The Security Profile Details screen appears.

15. Specify the Security Profile details, and click Next. Specify the following settings:

    - **Database User Account Name.**
    - **Database User Account Password.**
    - **Data Source Name.**
    - **Database Host Name.**
    - **Database Port Number.**
    - **Database Service Name.**

    The Gateway Registry screen appears.

16. Specify the Siebel Gateway registry configuration, and click Next. Specify the following settings:
    - **Security Profile.**
    - **Registry Port.**
    - **Primary Language.**
    - **Registry User Name.**
    - **Registry User Password.**

    The Application Interface Home screen appears.
17. Specify the Application Interface Home Location, and click Next.

    The Application Interface Details screen appears.
18. Specify the Application Interface details, and click Next. Specify the following settings:
    - **Application Interface Hostname.**
    - **Application Interface HTTPS Port.**
    - **Application Interface Username.**
    - **Application Interface Password.**
    - **Application Interface Truststore Name.**
    - **Application Interface Truststore Password.**

    The Summary screen appears.
19. In the Summary screen, review the information presented.

    This screen displays the location where Siebel CRM will be installed. Before you proceed, confirm that you have met all of the installation requirements, such as for disk space. Then do one of the following:
    - To begin installing Siebel CRM, click Install. You can also click Install after saving a response file.
    - To save a response file to use for a silent installation later, click Save Response File, and then save the file from the dialog box that appears. If you are not also installing at this time, then click Cancel after you save the response file. For information about performing silent installations, see *Installing Siebel CRM Using Silent Mode*.

    The Installation Progress screen appears. Siebel CRM is installed into the directory that you specified in Step 11.
20. When the Next button becomes available, the installation is complete. Click Next.

    The Finish Installation screen appears. This screen indicates whether the installation was successful and shows the location of the installer log file. You can access this file to review the installation results.
21. To close the installer, click Close.

    This migration installation of Siebel CRM is now finished.

    If you did not complete Enterprise Container Configuration or Application Interface Container Configuration, then you must run the Siebel CRM installer again and do so before you run Siebel Management Console.

Verifying Installation

Use the following procedure to verify that Siebel CRM installed successfully. For Microsoft Windows, use cmd. This task is part of *Installing Siebel CRM Using the User Interface*.

To verify the installation

1. Navigate to the directory represented by the `ORACLE_HOME` variable.
   For example, your top-level Siebel CRM installation directory on Microsoft Windows might be `c:\siebel`.
2. Run the following command:
   - (Windows) `set ORACLE_HOME=Oracle_Home`
   - (LINUX) `setenv ORACLE_HOME Oracle_Home`
3. Verify the version number of the installed Siebel CRM software. For more information, see *Verifying the Installation Version Number for Siebel CRM*.
4. If necessary, start the Siebel services and application containers. For more information, see *Stopping and Starting the Siebel Application Container*.
5. Start the Siebel application in the browser to make sure that the application starts successfully.
6. From the application-level menu, select the Help drop-down menu and open up the Technical Support dialog to check the version number displayed there.

Related Topics

Verifying Installation for Siebel CRM

Installing Siebel CRM Using Silent Mode

The Siebel CRM installer supports an optional *silent installation* (also known as *unattended installation*) process, which is an installation that has been preconfigured to require no user input when the installation program runs. You can save a response file in a regular installer session. Then you can run the installer at the command line with certain flags appended to the installer command to execute the response file. Together, these tasks constitute installing in silent mode.

You can perform silent installations for better performance when installing software on multiple computers. Or, you might perform silent installations if user input during an installation is not allowed in your environment, as might be the case in some secure environments. A silent installation prepackages all of the required parameters so that you only have to execute a command to perform the installation.

The silent installation process for the current release of Siebel CRM is similar to that for Siebel CRM 21.1 and earlier. You can use silent mode to perform any installation task by the Siebel CRM installer or to add languages or components to an existing installation.

For silent installation, Siebel CRM includes the following files. You can use these files or save and modify your own response files.

- enterpriseserver.resp
- siebelserver.resp

When you create a new response file from running the Siebel CRM installer, the file is created as siebelinstall.rsp and is located in the `SIEBEL_ROOT` installation directory.

> **CAUTION:** Silent installations provide no direct feedback or error notification. Therefore, you must test your settings in a development environment before system-wide deployment in a production environment. It is strongly recommended that you become thoroughly familiar with GUI installation and configuration for any Siebel CRM modules for which you intend to perform silent installation or configuration.

This topic includes the following information:

- *Guidelines for Installing Servers in Silent Mode*
- *Saving an Installation Response File*
- *Editing an Oracle-Provided Installation Response File*
- *Performing an Installation in Silent Mode*
- *Configuring Siebel CRM Using Silent Mode*

Related Topics

Installation Tasks for Siebel CRM

Installing Siebel CRM Using the User Interface

About Configuring Siebel CRM

Installing and Deploying Additional Languages

Verifying Installation for Siebel CRM

Requirements for Installing and Configuring Siebel CRM

Guidelines for Installing Servers in Silent Mode

Review the following guidelines for installing Siebel Enterprise Server or Siebel Application Interface in silent mode. This topic is part of *Installing Siebel CRM Using Silent Mode*.

- **Keep track of the response file results.** Because the silent installation session is controlled by the response file, you must keep careful track of the result each response file will yield when you perform a silent installation.

  For example, depending on your selections when you generated a response file for Siebel Enterprise Server, the silent installation performed with this file might install one or more of the following Siebel CRM server modules (subject to restrictions on installing modules together): Siebel Gateway, Siebel Server, Siebel Database Configuration Utilities, EAI Connector, or Siebel Application Interface.
- **Create the response file on a computer similar to the target computer.** When you run the installer as described in *Saving an Installation Response File*, installer validations are done as if you were performing a real installation. It is strongly recommended that you run the installer on a computer that has similar resources and configuration settings as the target computer on which the installation will be performed, as described in *Performing an Installation in Silent Mode*. Silent installations must meet all of the documented requirements, including those described in *Installation Tasks for Siebel CRM* and in *Requirements for Installing and Configuring Siebel CRM*.
- **Observe requirements for password encryption.** Password encryption is enabled by default for Siebel CRM. Therefore, any passwords you specify in the response file must have been encrypted using the EncryptString.jar utility, which is available in the following locations:

- In the Siebel installation image, EncryptString.jar is located in the `Disk1\stage\ext\jlib` directory.
- In the installed Siebel CRM software, EncryptString.jar is located in the `config\installer` directory.

To run the EncryptString.jar utility, use a command like the following:

```
Java.exe -jar <full_path_to_EncryptString.jar> <password to be encrypted>
```

- **Validate the installation.** Validate each installation after it has completed, as described in *Verifying Installation for Siebel CRM*.

To install Siebel CRM server modules and the specified Siebel languages using silent mode, perform the following tasks:

- *Saving an Installation Response File*
- *Editing an Oracle-Provided Installation Response File*
- *Performing an Installation in Silent Mode*

Saving an Installation Response File

Perform the following steps to save an installation response file to be used in a silent installation of Siebel CRM server modules. This task is part of *Installing Siebel CRM Using Silent Mode*.

> **Note:** If you are adding languages to an existing installation, then do not use this procedure. Instead, follow the procedure in *Editing an Oracle-Provided Installation Response File*.

To save an installation response file

1. Navigate to the Siebel image location where the installer is located.
2. Start the Siebel CRM installer.

   For information about starting this installer, see *Installation Tasks for Siebel CRM* and *Installing Siebel CRM Using the User Interface*.
3. Provide input to the installer wizard, as in a normal GUI installation.
4. In the Summary screen, click Save Response File.
5. Specify the location of the response file, for example, `C:\temp\sesinstall.rsp`, and save the file.

   A response file is generated that includes data that is derived from the input that you made during the installer session.
6. After the response file is saved, click Cancel.

   The installer session ends.

   In addition to saving the response file at the location that you specified at the command line, the installer creates the Siebel installation directory at the location that you specified in the installer session (for example, `C:\temp\sesinstall.rsp`). Although the Siebel CRM module is not installed (because you clicked Cancel), this directory contains a subdirectory called `inventory`, which includes information about the product inventory.

   For information about performing a Siebel Enterprise Server or Siebel Application Interface installation in silent mode, using the response file that you just generated, see *Performing an Installation in Silent Mode*.

Editing an Oracle-Provided Installation Response File

If you are adding languages to an existing Siebel CRM installation, then you must edit an Oracle-provided installation response file instead of saving a response file from the installer. For detailed information about the task of adding languages, see *Installing and Deploying Additional Languages*. This task is part of *Installing Siebel CRM Using Silent Mode*.

Performing an Installation in Silent Mode

Perform the following steps to perform a silent installation of Siebel CRM. This task is part of *Installing Siebel CRM Using Silent Mode*.

First, you must save an installation response file as described in *Saving an Installation Response File* or prepare an installation response file for adding languages as described in *Editing an Oracle-Provided Installation Response File*. Then you run the installer command as shown in the following procedure. For detailed information about the task of adding languages, see *Installing and Deploying Additional Languages*.

The installation commands shown in the following procedure run the installer in silent mode and specify the location of the response file to be executed. No installer screens are displayed, and no further user input is prompted for after the installer starts. The installation result depends on the content of the specified response file. The response file does not contain passwords. Consequently, you must specify the database table owner and database user password on the command line.

To perform an installation in silent mode

1. Navigate to the Siebel image location where the installer is located.
2. Place the response file that you previously generated or edited into the correct Siebel image directory for the Siebel CRM module (where the installer program is located), so that the file can be read when you run the installer.

   > **Note:** Always back up any existing valid response file before replacing it with a new file.

3. (Windows) Open a DOS command window, then run a command like the following to start the installer to install Siebel CRM:

   ```
   setup.bat -silent -responseFile path_to_installation_response_file
    -dp database_table_owner_password -sp database_user_password
   ```

   In this command:

   - `path_to_installation_response_file` is the full path and name of a response file to be executed (for example, `C:\temp\sesinstall.rsp`)
   - database_table_owner_password is the password for the database table owner
   - database_user_password is the password for the database user

4. (LINUX) Open a shell window, then run a command like the following to start the installer to install Siebel CRM:

   ```
   setup -silent -responseFile path_to_installation_response_file
    -dp database_table_owner_password -sp database_user_password
   ```

In this command:

- `path_to_installation_response_file` is the full path and name of a response file to be executed (for example, `/temp/sesinstall.rsp`)
- database_table_owner_password is the password for the database table owner
- database_user_password is the password for the database user

5. Validate the installation after it has completed.
   For more information, see *Verifying Installation for Siebel CRM*.

Configuring Siebel CRM Using Silent Mode

Silent configuration is an alternative to using the Siebel Management Console to configure the installed Siebel CRM server modules. Configuration and deployment of Siebel CRM, including the bootstrap process, can be done using differing methods, depending on the installed component. For information about performing Siebel Management Console configuration tasks, see *Configuring Siebel CRM Server Modules*. This task is part of *Installing Siebel CRM Using Silent Mode*.

The full Siebel Management Console (SMC) graphical user interface for ongoing management is provided across all configuration steps. These steps are based on REST or Java underlying the user interface. Individual components can also be managed by either REST or Java, as follows:

- REST-based steps can use the following:
  - Automation, postman, or other programmatic access.
  - Command-line interface, via HTTPIE (https://httpie.org) or Curl (https://curl.haxx.se). See command-line examples in this topic.
- Java-based steps can use silent mode using response file templates, provided as sample JSON files.

Common configuration steps and their implemented mechanisms are as follows:

1. REST: ZooKeeper/Gateway creation
2. REST: Security profile creation
3. REST: ZooKeeper/Gateway bootstrapping
4. Java: Enterprise configuration (can also be performed using REST)
5. Java: Siebel Server configuration (can also be performed using REST)
6. Java: Application Interface configuration

Sample JSON Files

Siebel CRM includes the following sample JSON files that you can modify as part of silent configuration. These files are located in the `SIEBEL_ROOT\config\sample_response` directory in your Siebel CRM installation.

- bootstrapCG.json
- GatewaySecurityProfile.json
- cginfo.json
- Alprofilecreate.json
- Alprofiledeploy.json

Detailed Steps

- Installing Siebel CRM in Silent Mode
- Validating Siebel Management Console
- Validating JSON Configuration Files
- Configuring Siebel Gateway Using Curl and JSON Files
- Configuring Siebel Enterprise and Siebel Server Using Response Files
- Refreshing SMC Configuration
- Configuring Siebel Application Interface

Installing Siebel CRM in Silent Mode

Example installation response can be found in the following directory in your Siebel CRM installation: `SIEBEL_ROOT\config\sample_response`. To install Siebel CRM in silent mode, perform the following general steps, as described in *Installing Siebel CRM Using Silent Mode*.

1. Obtain Siebel CRM 22.x installation media.
2. Validate the response files that you will use to install Siebel CRM using silent installation.
3. Install Siebel CRM using response files and silent installation.

Validating Siebel Management Console

1. Make sure Tomcat is up and running with HTTPS port (`AI_ROOT/applicationcontainer/logs/catalina*.log`).

   Sample log:

   ```
   19-Jan-2021 19:31:42.106 INFO [main] org.apache.coyote.AbstractProtocol.start Starting ProtocolHandler
    [http-nio-9002]
   19-Jan-2021 19:31:42.154 INFO [main] org.apache.coyote.AbstractProtocol.start Starting ProtocolHandler
    [https-jsse-nio-9001]
   19-Jan-2021 19:31:42.169 INFO [main] org.apache.catalina.startup.Catalina.start Server startup in 41568
    ms
   ```

2. Log in to SMC:

   ```
   https://<AI_HOST>:<AI_PORT>/siebel/smc
   ```

3. Make sure the pop-up for Gateway Host:Port is visible.

Validating JSON Configuration Files

1. Using a Windows or LINUX computer where Curl is installed, copy the following JSON files to the same location where Curl is enabled:

   - cginfo.json
   - GatewaySecurityProfile.json
   - bootstrapCG.json

   For more information about these files, see the sample JSON files listed in this topic.
2. Modify the JSON files according to your environment.

Configuring Siebel Gateway Using Curl and JSON Files

Run Curl commands similar to the following, representing different phases of configuration for Siebel Gateway and Security profiles:

1. Siebel Gateway:

```
curl -u <AI_USER>:<AI_PASSWORD> -i -X POST -H "Content-Type:application/json" "https://
<AI_HOST>:<AI_PORT>/siebel/v1.0/cginfo" -d@cginfo.json -k
```

2. Security:

```
curl -u <AI_USER>:<AI_PASSWORD> -i -X POST -H "Content-Type:application/json" "https://
<AI_HOST>:<AI_PORT>/siebel/v1.0/cloudgateway/GatewaySecurityProfile" -d@GatewaySecurityProfile.json -k
```

3. Bootstrap:

```
curl -u <DB_USER>:<DB_USERPASSWORD> -i -X POST -H "Content-Type:application/json" "https://
<AI_HOST>:<AI_PORT>/siebel/v1.0/cloudgateway/bootstrapCG" -d@bootstrapCG.json -k
```

Error output for Curl commands shows HTTP 200 OK to indicate success. Review output in detail for any issues. Once the preceding steps are successful, validate SMC. It should show the security profile created.

Example GatewaySecurityProfile.json Configured for Database Authentication

To configure database authentication instead of LDAP authentication, update the GatewaySecurityProfile.json file accordingly before performing the configuration steps. Adapt this content as appropriate for your deployment requirements:

```
{
 "Profile": {
 "ProfileName": "OracleProfile"
 },
 "SecurityConfigParams": {
 "DataSources": [{
 "Name": "Oracle",
 "Type": "DB",
 "Host": "xxxxxx",
 "Port": xxxx,
 "SqlStyle": "Oracle",
 "Endpoint": "xxxx",
 "TableOwner": "xxxx",
 "HashUserPwd":false,
 "HashAlgorithm":"SHA1",
 "CRC":"0"
 }],
 "SecAdptName": "DBSecAdpt",
 "SecAdptMode": "DB",
 "NSAdminRole": ["Siebel Administrator"],
 "TestUserName": "xxxx",
 "TestUserPwd": "xxxxx",
 "DBSecurityAdapterDataSource": "Oracle",
 "DBSecurityAdapterPropagateChange": true
 }
}
```

Configuring Siebel Enterprise and Siebel Server Using Response Files

To create response files, first configure a Siebel environment manually using SMC, then create response files that you can use for configuring additional environments.

1. Copy the response files (RESP files) to the `c:\siebel\ses\config` directory.
2. Modify the response files for the Siebel Enterprise and Siebel Server with required parameters.
3. Open a command prompt. Run the following command:

   ```
   set JAVA_HOME=$ses/jre
   ```

4. Go to `c:\siebel\ses\config` and run the following command:

   Windows:

   ```
   config.bat -mode enterprise -responseFile enterpriseserver.resp -verbose -skipvalidation
   ```

   LINUX:

   ```
   config.sh -mode enterprise -responseFile enterpriseserver.resp -verbose -skipvalidation
   ```

   If this step fails, troubleshoot and run the step again.
5. For Siebel Server configuration, run the following command:

   Windows:

   ```
   config.bat -mode siebsrvr -responseFile siebelserver.resp -verbose -skipvalidation
   ```

   LINUX:

   ```
   config.sh -mode siebsrvr -responseFile siebelserver.resp -verbose -skipvalidation
   ```

6. Once successful, validate that the Siebel Server service is started from services.
7. Connect to srvrmgr and verify that the Siebel Server is up and running.

Refreshing SMC Configuration

Because you configured Siebel Enterprise and Siebel Server with response files, these updates are not reflected in SMC until you refresh the SMC configuration. Follow these steps to refresh the configuration:

1. Connect to SMC with a URL like the following:

   https://<AI_HOST>:<AI_PORT>/siebel/smc
2. Go to Settings. Click Refresh Configuration.
3. Go to Deployment. You can see that the Enterprise and Siebel Server profiles are migrated.

Configuring Siebel Application Interface

1. Using Curl, update the JSON files for profile creation and deployment, as follows:
   - Profile creation command:

     ```
     curl -u <DB_USER>:<DB_USERPASSWORD> -i -X POST -H "Content-Type:application/json"
      "https://"$RESTHOSTNAME":"$portSAIHTTPS"/siebel/v1.0/cloudgateway/profiles/swsm/" -
     d@AIprofilesiacreate.json -k
     ```

- Profile deployment command:

```
curl -u <DB_USER>:<DB_USERPASSWORD> -i -X POST -H "Content-Type: application/json"
 "https://"$RESTHOSTNAME":"$portSAIHTTPS"/siebel/v1.0/cloudgateway/deployments/swsm/" -
d@AIprofiledeploy.json -k
```

2. Validate the Application Interface profile details from SMC.

Additional Tasks for Migration Installations

Certain product changes that were made in Siebel CRM spanning several releases can affect migration installations for existing customers. All customers performing migration installations must review the information in this topic before installing, to make sure that they can successfully migrate all custom files and settings to the current release and to avoid any of the issues described here. Some migration tasks previously documented for Siebel CRM 16.0 might not need to be done again for customers migrating from that release. The migration installation case applies only for migrations from Siebel CRM 16.x or earlier.

These product changes affect some of the files and directories of installations of Siebel Server, Siebel Application Interface, Siebel Web Client, and Siebel Tools. Also affected are some of the configuration settings for these modules. Many of the product changes described in this topic were made as part of consolidating support for Siebel Open UI and desupporting high interactivity and standard interactivity.

This topic describes specific product changes, describes how these changes might affect migration installations, and provides steps you can take to avoid any issues and complete the migration successfully. Some of the tasks you perform before installation, but most of them you would perform after installation. Additional migration considerations are also provided.

This topic includes the following information:

- *Migration Issues and Solutions for Siebel CRM 17.0 and Later*
- *Updating the Siebel Server System Service and Server Components to Use AES Password Encryption*

Related Topics

Installation Tasks for Siebel CRM

About Configuring Siebel CRM

About Installing Siebel Web Client or Siebel Tools

Installing and Using Oracle Database SE2 for the Local Database

Uninstalling Siebel CRM

Process of Removing Configuration Data

Migration Issues and Solutions for Siebel CRM 17.0 and Later

Many product and support changes for Siebel CRM 17.0 and later releases mean you need to perform certain migration-related tasks. Issues and possible solutions are provided for some items. Review all items before proceeding. This list isn't comprehensive.

Many of the tasks are interrelated and aren't necessarily presented in the order in which you'd perform them. Various dependencies apply that might not be fully explained. Some of these issues apply only to migrations from releases before Siebel CRM 16.0.

This topic is part of *Additional Tasks for Migration Installations*.

- **Sync local databases before migrating or doing other premigration tasks.** Siebel Mobile Web Client users might need to sync their local database changes with the server, through Siebel Remote, as described in *Siebel Remote and Replication Manager Administration Guide* . Or, Siebel Tools developer users might need to check in any changes made to objects in the existing local database, as described in *Using Siebel Tools* for earlier releases of Siebel Tools.
- **Remove the Siebel Web Server Extension (SWSE) configuration before migrating, and then configure Siebel Application Interface after migrating.** In Siebel CRM 16.0, the virtual directories, which formerly mapped to `public\lang_code` (such as ENU), were mapped directly to `public`. Virtual directories no longer apply and are replaced by application configurations. In the current release, these configurations map to `applicationcontainer_external\siebelwebroot` in the Siebel Application Interface installation. When you install the Siebel Application Interface as a migration installation, the existing virtual directories on the Web server that were configured for SWSE in a prior release are not migrated. After all of the migration installations are complete, you must configure the Siebel Application Interface to create the application configurations. For more information, see *Installation Tasks for Siebel CRM* and *About Configuring Siebel CRM*.

  > **CAUTION:** You must remove the configuration for the SWSE before you install Siebel CRM 17.0 as a migration installation, as described in *Process of Removing Configuration Data*.

- **When data encryption is enabled, you must back up the original key file (keyfile.bin) before performing a migration installation for Siebel Enterprise Server, and then copy it back after migrating.** For more information, see *General Requirements for Installing and Configuring Siebel Enterprise Components*.

  > **CAUTION:** If data migration is enabled, then the migration installation overwrites your existing key file. If you have not backed up your existing key file and copied it back after the migration, then the encrypted columns will be inaccessible after the migration.

- **Preserve premigration installation.** You might want to preserve your premigration installations, in case you need to restore your prior version after having done the migration installations and made the manual migration changes described here. It is recommended that you manage your migration tasks to make sure that you have not permanently deleted, moved, or renamed files or directories in these installations. Also keep track of changed requirements, so that you can restore the overall environment for your prior release. For more information, see *Uninstalling Siebel CRM*.
- **Move files and directories or delete unnecessary files.** Depending on the release you are migrating from and on your requirements, you might need to copy some files or directories from your existing installation into your new migrated installation, or you might need to delete some unnecessary files and directories from your new migrated installation. For example:
  - **The webtempl directory on Siebel Server.** As of Siebel CRM 17.0, the `SIEBSRVR_ROOT\webtempl` directory is not part of the Siebel Server installation and is not migrated. The Siebel Web templates are now located

in the Siebel database. Custom Siebel Web templates for Siebel Open UI migrate into the database when you run Incremental Repository Merge, as described in *Siebel Database Upgrade Guide* .
In order for this migration to succeed, you must copy all of the applicable custom Siebel Web template files into a directory. Then, when you run Incremental Repository Merge and the Database Configuration Wizard prompts for the Web Templates Directory, specify this new directory location.

- If you are migrating from Siebel CRM 16.x, then the applicable Siebel Web template files to copy are those located in the `SIEBSRVR_ROOT\webtempl\custom` subdirectory in your prior installation.
- If you are migrating from a release prior to Siebel CRM 16.0, then the applicable Siebel Web template files to copy are those located in the `SIEBSRVR_ROOT\webtempl\ouiwebtempl\custom` subdirectory in your prior installation.
  For Siebel Web Client and Siebel Tools installations, the `webtempl` directory has also been removed and Siebel Web templates are now provided in the Siebel database. A newly extracted local database for Siebel Mobile Web Client also includes your custom Siebel Web templates.

- **The webmaster directory on Siebel Server.** As of Siebel CRM 16.0, the `SIEBSRVR_ROOT\webmaster` directory on the Siebel Server has been removed. If you are migrating from a release prior to Siebel CRM 16.0, then you can delete this directory from your migrated Siebel Server installation.
- **The public directory on Siebel Web Server Extension.** As of Siebel CRM 17.0, the `SWSE_ROOT\public` directory is not part of the Siebel Application Interface installation. In the current release, the equivalent location is `SIEBEL_AI_ROOT\applicationcontainer_external\siebelwebroot`. Files and directories are not automatically migrated for Siebel Application Interface. If you require any of the files from the `public` directory in your prior installation of SWSE, then you can copy them manually after completing installation and configuration tasks.
  Before you copy files, you must take into account the following changes that occurred in the structure of the `public` directory in Siebel CRM 16.0:
  - `public\lang_code\build_number` , where lang_code is an installed Siebel language (such as ENU) and build_number is one of the build numbers for the installed Siebel software. The `applet` subdirectory was removed. The `scripts` subdirectory moved under `public`. And the `build_number` directory was removed.
  - `public\lang_code\files`. This directory moved under `public`.
  - `public\lang_code\fonts`. This directory moved under `public`.
  - `public\lang_code\htmltemplates`. This directory moved under `public`.
  - `public\lang_code\images`. This directory moved under `public`.
  - `public\lang_code\webeditor`. This directory was removed.
  - `public\lang_code\default.htm`. This file moved under `public`.
  - `public\lang_code\blank.htm`. This file moved under `public`.
  - `public\lang_code\wait.htm`. This file was removed.
  - `public\lang_code\*.pcd`. These files were removed.
  - `public\lang_code\*.manifest`. The variable string %BuildNumber% was removed from the .tmanifest files, from which the .manifest files are generated.

  > **Note:** The Siebel application virtual directories, which formerly mapped to the `public\lang_code` directory on SWSE, are obsolete as of Siebel CRM 17.0. In the current release, the application configurations for Siebel Application Interface map to `SIEBEL_AI_ROOT\applicationcontainer_external\siebelwebroot`.

- **Move files and directories: Files and directories on Siebel Web Client and Siebel Tools have changed relative to equivalent locations for releases prior to Siebel CRM 16.0.** Similar changes apply for migration

installations of Siebel Web Client and Siebel Tools, as described above for migration installations of Siebel Application Interface. However, the files are copied into the new migration installations of Siebel Tools. Note also that Siebel Web Client still uses the `public` directory, as in Siebel CRM 16.0.

Issue (for migrations from release prior to Siebel CRM 16.0). After a migration installation of Siebel Web Client or Siebel Tools (from a release prior to Siebel CRM 16.0), duplicate directories will exist, because some of the directories will be in new locations relative to those seen for the files and directories that migrated from the prior release. Your custom files will be in the wrong locations and cannot be used until you have moved them to the new locations.

Solution. Reorganize the files and directories in `SIEBEL_CLIENT_ROOT\public` and `SIEBEL_TOOLS_ROOT\public` on the migrated installations to incorporate the migrated custom files that you require and to conform to the changes described earlier in this topic for Siebel Application Interface. Finally, remove all extraneous files and directories in `SIEBEL_CLIENT_ROOT\public` and `SIEBEL_TOOLS_ROOT\public`.

- **Upgrading the database client.** In a migration installation case, the database client software is installed for use with the current release of Siebel CRM automatically, and supported with databases. Oracle Database Client is automatically installed with Siebel Enterprise Components. For more information, see *About Configuring Siebel CRM*.

  Installing Siebel Web Client also installs Oracle Database SE2 and Oracle Database Client, as noted in *Installing and Using Oracle Database SE2 for the Local Database* and other topics.

- **Edit configuration files for Siebel Web Client.** After doing a migration installation for Siebel Web Client from Siebel CRM 16.x or earlier, you might need to edit all applicable migrated configuration files, such as uagent.cfg for Siebel Call Center. For example:

  For a migration from Siebel CRM 16.x or earlier, in the [Siebel] section of each applicable configuration file, the existing WebClientSiteDir parameter value will include a language element, which is no longer needed. For example, for a value like `c:\Siebel\Client\public\enu`, update the parameter value to `c:\Siebel\Client\public`.

- **Remove obsolete parameters.** Several parameters for server components or in configuration files are obsolete as of Siebel CRM 17.0. To reduce confusion after migration installations, you might choose to delete some of the parameters that no longer apply, such as HighInteractivity and EnableOpenUI (which are obsolete as of Siebel CRM 16.0). For more information about obsolete parameters, see *Siebel System Administration Guide* .

- **Reset passwords on the Siebel Gateway and on the Siebel Server.** After doing a migration installation from Siebel CRM 16.x or earlier, you must reset any passwords on the Siebel Gateway that were previously encrypted using encryption other than AES. Such passwords are now encrypted using AES. For more information about reencrypting these passwords, see *Siebel Security Guide* . Furthermore, the Siebel Server system service and server components do not work after a migration installation until you have updated them to use AES password encryption. Make these changes in coordination, as described in *Updating the Siebel Server System Service and Server Components to Use AES Password Encryption*.

Updating the Siebel Server System Service and Server Components to Use AES Password Encryption

The Siebel Server system service and server components do not work after a migration installation until you have performed steps to update them to use AES password encryption, as described in the following procedure. Most of the command examples in the procedure are for LINUX operating systems, but the same issue applies on Microsoft Windows.

This task is part of *Additional Tasks for Migration Installations*.

To update the Siebel Server system service and server components to use AES password encryption

1. Unset any SIEBEL* environment variables.
2. Source the siebenv script from the Siebel Gateway installation directory, as follows:

   ```
   ./siebenv.sh
   ```

3. Start the Siebel Gateway and make sure that it is running.
4. Write down the old encrypted password from the current siebns.dat file. For example:

   ```
   [/enterprises/esia81/parameters/Password]
   Persistence=full
   Type=string
   Value="9ntkUOUf"
   Length=16
   ```

5. Run a command like the following, using the old encrypted password value:

   ```
   SIEBEL_ROOT/siebsrvr/lib/spu 9ntkUOUf
   ```

   This command obtains a reencrypted value for the password, such as in the following output:

   ```
   ENPVR6S/HKgBncoAAA==
   ```

6. Delete the current svc* and osdf* files from the `SIEBSRVR_ROOT/sys` directory.
7. Source the siebenv script from the `SIEBSRVR_ROOT` directory.
8. Change directory to the `SIEBSRVR_ROOT/bin` directory and then run a command like the following:

   ```
   siebctl -S siebsrvr -i esia81:srvr1 -a -g "-g localhost:2320 -e esia81 -s srvr1
   -u SADMIN -ep ENPVR6S/HKgBncoAAA=="
   ```

   Note that the new encrypted string from Step 5 is used.
9. Run the following command using the old encrypted password value, like the following:

   ```
   $SIEBEL_ROOT/siebsrvr/lib/gpu -g localhost:2320 -e esia81 -u SADMIN -p 9ntkUOUf
   ```

10. Customers who have overridden the user name or password at the component definition level must change the passwords again through srvrmgr in order to use the new encrypted password value. For example:

    ```
    change param password=<pwd_value> for compdef <comp_name>
    change param password=<pwd_value> for comp <comp_name> server <server_name>
    ```

11. Update old encrypted passwords on the Siebel Gateway to use AES encryption.

    For more information, see *Siebel Security Guide* .
12. Restart both the Siebel Gateway and the Siebel Server.

4 Configuring Siebel CRM Server Modules

Configuring Siebel CRM Server Modules

This chapter explains how to use Siebel Management Console to configure all of the Siebel CRM modules that you can install using the Siebel CRM installer. It includes the following topics:

- *About Configuring Siebel CRM*
- *Running the Siebel Management Console*
- *Installing the Siebel Database*
- *Configuring the Siebel Gateway and Security*
- *Configuring the Siebel Enterprise*
- *Configuring the Siebel Server*
- *Configuring the Siebel Application Interface*
- *Configuring the Siebel Gateway Cluster*
- *Configuring the Siebel Migration Application*
- *Configuring the Siebel Deployment*
- *Managing the Siebel Deployment*
- *Accessibility for Siebel Management Console*

For information about configuring Siebel CRM server modules using silent mode, see *Configuring Siebel CRM Using Silent Mode*.

About Configuring Siebel CRM

After performing a new installation of Siebel Enterprise Server modules or Siebel Application Interface, you must configure the appropriate modules by using the Siebel Management Console, as described in this chapter.

The Siebel Management Console, a Web-based application that you run on the Siebel Application Interface, supports configuration tasks for the following:

- Security
- Siebel Gateway
- Siebel Gateway Cluster
- Siebel Enterprise
- Siebel Server
- Siebel Application Interface
- Siebel Migration application

Siebel Management Console also provides the ability to configure and manage deployed server entities, similar to the functionality in the Administration - Server Configuration and Administration - Server Management screens in the Siebel application.

In most cases, you perform the configuration in a two-step process for each entity that you are configuring:

- Creating a profile for the type of entity you are configuring
- Deploying the profile into your Siebel deployment

A profile is a collection of configuration settings. You can create as many profiles as you require, and deploy these profiles according to your requirements, as appropriate for your physical installed Siebel software and your authentication methods. You can deploy profiles to multiple installed instances of Siebel Server or Siebel Application Interface, for example. When you create a profile, you must use a unique name in order to successfully create the profile.

Most of the configurable entities correspond to physical installed modules (Siebel Gateway, Siebel Server, or Siebel Application Interface). Some entities are provided with other modules. For example, security adapters are provided with the Siebel Server, and the Siebel Migration application is provided with the Siebel Application Interface installation. The Siebel Enterprise is a logical entity that you must deploy for each Siebel deployment. The Siebel Gateway Cluster is also a logical entity that you can optionally configure for multiple installed Siebel Gateway nodes.

You must configure and deploy at least the Siebel Gateway, Siebel Enterprise, Siebel Server, and Siebel Application Interface in every Siebel deployment.

After you configure the Siebel Gateway, you can create profiles in any sequence. Restrictions apply to the sequence in which you deploy these profiles in your Siebel deployment. In general, the configuration sequence for a new deployment corresponds to the sequence of the configuration topics in this chapter.

> **Note:** For a migration installation case, you must install Siebel Application Interface first and specify information such as application container port numbers. You then provide some of this data as input for the migration installation of Siebel Gateway, which allows the configurations to be migrated. After the migration installations, you do not configure the Siebel Gateway, Siebel Enterprise, or Siebel Server using the Siebel Management Console. However, you must then configure the Siebel Application Interface. Additional configuration requirements apply for migration installations, as described in *Configuration Requirements for Migration Installations*.

The Siebel Management Console uses the Siebel Gateway registry to store configuration data. Consequently, the first configuration task that you must perform when you run the Siebel Management Console is configuring the Siebel Gateway itself, and its registry. After doing this, you can create and deploy profiles for all other applicable modules in your Siebel deployment.

> **Note:** It is strongly recommended to back up the Siebel Gateway registry periodically for safety reasons, so that you will be able to restore the Siebel Gateway registry from a backup if something goes wrong. For more information, see *Siebel System Administration Guide* .

(Windows) After configuration, the Siebel system services start automatically. You can use the Windows Control Panel to change whether a Siebel system service starts automatically or manually, or to change the user name or password under which the system service runs.

(UNIX) After configuration, you might want to configure the Siebel system services to start automatically.

> **Note:** You cannot operate any Siebel CRM modules for your deployment until you have configured and deployed them. You must be running the Siebel Gateway in order to configure and deploy any other entities.

Other Siebel Management Console Activities

You can also use the Siebel Management Console to remove configuration data for deployed entities, which is a necessary part of uninstalling the Siebel CRM software. Restrictions apply to the sequence in which you can remove configuration data for deployed modules. For more information, see *Process of Removing Configuration Data*.

To modify some configuration settings for an existing Siebel Enterprise or Siebel Server, or a Siebel Server component, you can optionally use the functionality in the Configuration screen in Siebel Management Console, as described in *Configuring the Siebel Deployment*. Alternatively, you can use the Administration - Server Configuration screen in the Siebel application or use Siebel Server Manager, as described in *Siebel System Administration Guide* . Depending on the specific configuration updates that you make, you might have to restart the Siebel Server before the changes take effect. The Siebel Management Console allows you to refresh the configuration data displayed so that it reflects the latest updates for the deployed Siebel Enterprise or Siebel Server.

In addition, you can manage server modules in the Siebel CRM deployment by using the Management screen in the Siebel Management Console, as described in *Managing the Siebel Deployment*. Alternatively, you can use the Administration - Server Management screen in the Siebel application or use Siebel Server Manager, as described in *Siebel System Administration Guide* .

This topic includes the following information:

- *Where to Find Configuration Tasks*
- *Configuration Requirements for Migration Installations*

Where to Find Configuration Tasks

The configuration tasks for Siebel CRM are described in the following locations, as shown in the following table. This topic is part of *About Configuring Siebel CRM*.

Siebel CRM Module to Configure	Where Documented
Siebel database	*Installing the Siebel Database* **Note:** Installation of a new Siebel database is still performed using the Database Configuration Wizard.
Security	*Configuring the Siebel Gateway and Security* and *Siebel Security Guide*
Siebel Gateway	*Configuring the Siebel Gateway and Security*
Siebel Gateway Cluster	*Configuring the Siebel Gateway Cluster*
Siebel Enterprise	*Configuring the Siebel Enterprise*
Siebel Server	*Configuring the Siebel Server*
Siebel Application Interface	*Configuring the Siebel Application Interface*

Siebel CRM Module to Configure	Where Documented
Siebel Migration application	*Configuring the Siebel Migration Application* and *Siebel Database Upgrade Guide*
Siebel deployment (optional post-deployment configuration tasks)	*Configuring the Siebel Deployment* and *Siebel System Administration Guide*
Siebel deployment (optional post-deployment management tasks)	*Managing the Siebel Deployment* and *Siebel System Administration Guide*

Configuration Requirements for Migration Installations

The migration installation case has particular requirements that relate to configurations. For your installed Siebel CRM software for a prior release (Siebel CRM 16.x or earlier), do the following, in the order shown. This topic is part of *About Configuring Siebel CRM*.

1. Review all relevant documentation. See *Before You Install Siebel CRM* and *Installation Tasks for Siebel CRM*.
2. Perform any necessary premigration tasks, as noted in *Additional Tasks for Migration Installations*.

   > **CAUTION:** In particular, you must remove the existing Siebel Web Server Extension (SWSE) configuration on the Web server computer where you installed SWSE for the prior release. Do this before you perform the migration installation of Siebel Application Interface. Otherwise, the virtual directories cannot be cleaned up adequately. For more information, see *Removing the Siebel Application Interface Configuration*.

3. Install Siebel Application Interface as a migration installation for your existing installation of SWSE, as described in *Installing Siebel CRM in a Migration Installation*.
4. Install Siebel Enterprise Server as a migration installation, as described in *Installing Siebel CRM in a Migration Installation*.

   > **Note:** For the Siebel Enterprise Server that includes the Siebel Gateway, when you perform the migration installation, you provide input that configures the Siebel Gateway and allows the existing configurations for the Siebel Enterprise and Siebel Servers to migrate to the new Siebel Gateway. This instance is considered the primary Siebel Gateway if you deploy Siebel Gateway clustering.

5. (Optional) Install Siebel Enterprise Server as a new installation, and install any other Siebel CRM modules that you require, as described in *Installing Siebel CRM in a New Installation*. For example, install Siebel Gateway (if you are using Siebel Gateway clustering on multiple nodes).
6. Perform any necessary postinstallation tasks, as noted in *Additional Tasks for Migration Installations*. For example, do the following:
   - Update the database client installation path in the defined environment variables.
   - Update files and directories on Siebel Enterprise Server and Siebel Application Interface (where necessary).
7. Run Incremental Repository Merge, as described in *Siebel Database Upgrade Guide* .

8. Run the Siebel Management Console, as described in *Starting the Siebel Management Console*. See also *About Configuring Siebel CRM*.
    - (Optional) Create a profile for the Siebel Gateway cluster, if you have installed Siebel Gateway on multiple nodes for this purpose, and then deploy this profile to deploy the cluster. For more information, see *Configuring the Siebel Gateway Cluster*.
    - Create a profile for Siebel Application Interface, and then deploy this profile to the installed Siebel Application Interface. For more information, see *Configuring the Siebel Application Interface*.
    - Configure any new Siebel CRM modules that you installed, including those noted in a prior step, or configure the Siebel Migration application.

Related Topics

Installation-Related Changes in Siebel CRM

Siebel CRM Download and Installation

Installation Tasks for Siebel CRM

Additional Tasks for Migration Installations

Running the Siebel Management Console

Installing the Siebel Database on the RDBMS

Additional Postinstallation and Configuration Tasks

Requirements for Installing and Configuring Siebel CRM

Process of Removing Configuration Data

Related Books

Siebel Database Upgrade Guide

Running the Siebel Management Console

This topic describes how to run the Siebel Management Console. For more information, see *About Configuring Siebel CRM*. It includes the following information:

- *Starting the Siebel Management Console*
- *Overview of Siebel Management Console*
- *Siebel Management Console Screens*
- *Supported Activities in Siebel Management Console*
- *Using Safe Mode for Siebel Management Console*
- *Securing Siebel Management Console Deployments*

See also *Accessibility for Siebel Management Console*.

Starting the Siebel Management Console

Use the following instructions to start the Siebel Management Console. This task is part of *Running the Siebel Management Console*.

To start the Siebel Management Console

1. Make sure that you have installed all the necessary Siebel CRM software and performed any required postinstallation steps.

   Depending on your authentication method, you might also have had to install or upgrade the Siebel database before you run the Siebel Management Console.
2. Review the information in *About Configuring Siebel CRM*.
3. Open a browser window and enter a URL like the following:

   https://SiebelApplicationInterfaceHost.FQDN:Port/siebel/smc

   In this URL:

   - SiebelApplicationInterfaceHost is the host name for the installed Siebel Application Interface.
   - FQDN is the fully qualified domain name for your deployment. An example is example.com.

     > **Note:** It is strongly recommended to use the fully qualified domain name (FQDN) when you enter a URL, because security certificates are generally obtained based on fully qualified domain name.

   - Port is the HTTPS redirect port number for the installed Siebel Application Interface. You specified this HTTPS redirect port during Siebel Application Interface installation.
4. On the login page that appears, enter the following information, and then click Login.

   - **User ID.** The first time you log in, you must use the credentials that you specified in the Application Interface Authentication screen when you installed Siebel Application Interface, as described in *Installing Siebel CRM in a New Installation*. Otherwise, use valid credentials for the authentication method in effect. For example, you might log in using the SADMIN account.
   - **Password.** The password for this user ID.
   - **Language.** The language in which to run the Siebel Management Console. You can select any of the languages that were installed on the Siebel Application Interface.

   > **Note:** After you have logged in to the Siebel Management Console for the first time and configured security for the Siebel Gateway, then you must log in to the Siebel Management Console again before you can proceed with additional configuration tasks. The Siebel Gateway must be running. For information about specific configuration tasks, see *Configuring the Siebel Gateway and Security*.

   If you have already configured the Siebel Gateway, or if you performed a migration installation of the Siebel Gateway instead of a new installation, then go to the topic for the configuration task that you are performing, as shown in the table in *Where to Find Configuration Tasks*.

To start Siebel Management Console in safe mode

1. Open a browser window and enter a URL like the following:

   SiebelApplicationInterfaceHost.FQDN:Port/siebel/smc/safemode.html

   The variable elements in this URL are the same as those described for starting Siebel Management Console in the normal manner.
2. Log in to Siebel Management Console using the safe mode user credentials.

   > **Note:** Use safe mode only when strictly necessary, as described in *Using Safe Mode for Siebel Management Console*. See also *Siebel Security Guide* .

3. Update the Siebel Gateway security profile, such as to change the data source information.
4. Save the Siebel Gateway security profile.

   When you save the profile, you are automatically logged out of safe mode and redirected to the normal login screen for Siebel Management Console.
5. Where necessary, log in to Siebel Management Console again in the usual way to perform configuration or other tasks.

Related Topics

About Configuring Siebel CRM

Overview of Siebel Management Console

Supported Activities in Siebel Management Console

Using Safe Mode for Siebel Management Console

Configuring the Siebel Gateway and Security

Accessibility for Siebel Management Console

Related Books

Siebel Security Guide

Overview of Siebel Management Console

Siebel Management Console is a Web-based application that you run on the Siebel Application Interface. The configuration tasks that you perform in this program use the graphical user interface shown in the figures. This topic is part of *Running the Siebel Management Console*.

> **Note:** Where possible, it is strongly recommended to explore this user interface as much as possible before performing important configuration tasks for any Siebel environment. For many tasks, multiple ways of performing those tasks are provided.

The Siebel Management Console, which displays this name in its title bar, includes user interface features such as menus, icons, lists and forms, and a hierarchical display of your Siebel deployment. The options and defaults that the

Management Console displays depend on which module you are configuring and on prior selections. For each profile configuration task, the settings you see are also displayed in a table.

The Siebel Management Console performs validation on the input that you provide. For example, you cannot proceed to the next screen unless you have provided values for all required fields. Pointing to items displayed in the console displays tooltips for many of the selections, which supplement the information provided in this guide.

Sample configurations are provided for most types of configuration profiles. As a shortcut for creating profiles, you can select these sample profiles and review their settings. As appropriate, you can copy and modify these profiles for your own use.

Click each section to review its options. When you are creating a new profile, you must specify settings for all required options, clicking Next where necessary to display options in subsequent sections. When you are reviewing an existing profile, the Next button is not available. However, for applicable profile types, you can view additional sections by clicking the next segment of the multipart horizontal bar.

The Siebel Management Console, which you run in the browser from the Siebel Application Interface installation, uses RESTful services and application containers to communicate with the physical Siebel CRM modules that you are configuring or that participate in the configuration process.

The Siebel Management Console primarily supports the initial configuration of Siebel CRM. In addition to creating profiles and deploying them to your initial Siebel deployment, you can configure modules you install subsequently to add to your existing deployment. You can also modify profiles (subject to limitations), clone profiles, compare profiles, and delete profiles.

You can modify the configuration of existing deployment items in Siebel Management Console, as follows:

- For Siebel Application Interface, the profile is in a read-write state after you deploy it. You can update configuration settings and save the profile to propagate the updates to the deployed Siebel Application Interface.
- For all other deployment items, the profile that you had used to configure the entity is in a read-only state after deployment. To modify the deployment item, you must remove it, modify the applicable profile, and then redeploy the profile to a new deployment item. For more information about deleting a deployment item, see *Process of Removing Configuration Data*.
- Deployment items in the Saved or Staging states can be modified. For other than Siebel Application Interface, deployments in any other state are not editable and cannot be modified other than as described in the previous paragraph.

Also note the following about the display of configuration data in Siebel Management Console:

- Restarting the Siebel Application Interface refreshes all profile data.
- If the deployment is migrated successfully by the Siebel Migration application, then, when you refresh the configuration, the state changes to Migrated.

Further configuration and management capabilities of Siebel Management Console are available as described in *Configuring the Siebel Deployment* and *Managing the Siebel Deployment*. For information about accessibility features for Siebel Management Console, see *Accessibility for Siebel Management Console*.

Siebel Management Console Screens

The screens within the Siebel Management Console are described here. This topic is part of *Running the Siebel Management Console*.

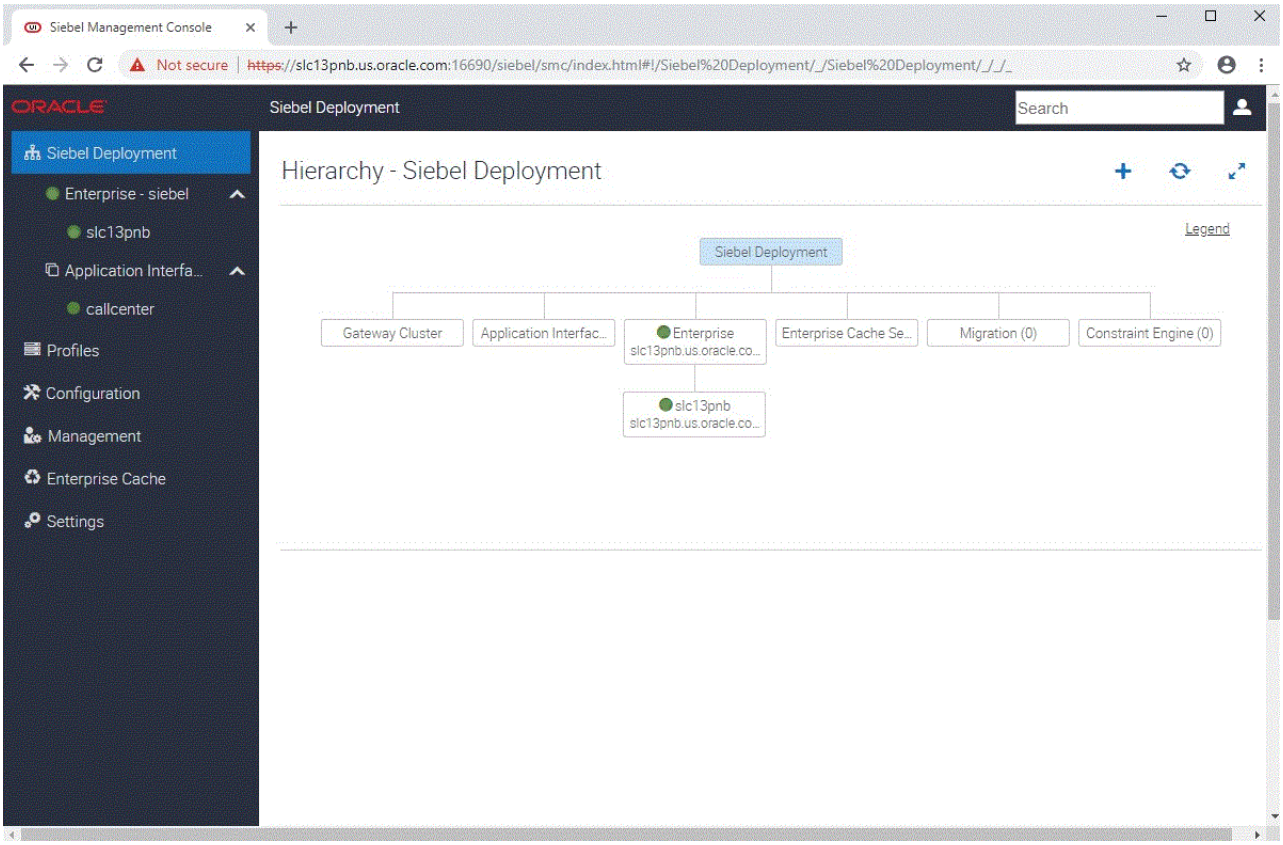
You can move from one screen to another by choosing a different selection from the navigation menu in the side panel. Alternatively, in some views of the Siebel Deployment, Profiles, Configuration, and Management screens, you can move to another screen by clicking an icon representing that screen, while also preserving context. For example, if you are configuring a Siebel Server, then clicking the icon for the Siebel Deployment screen navigates to that screen, with the same Siebel Server selected.

See also *Supported Activities in Siebel Management Console*.

Siebel Deployment Screen

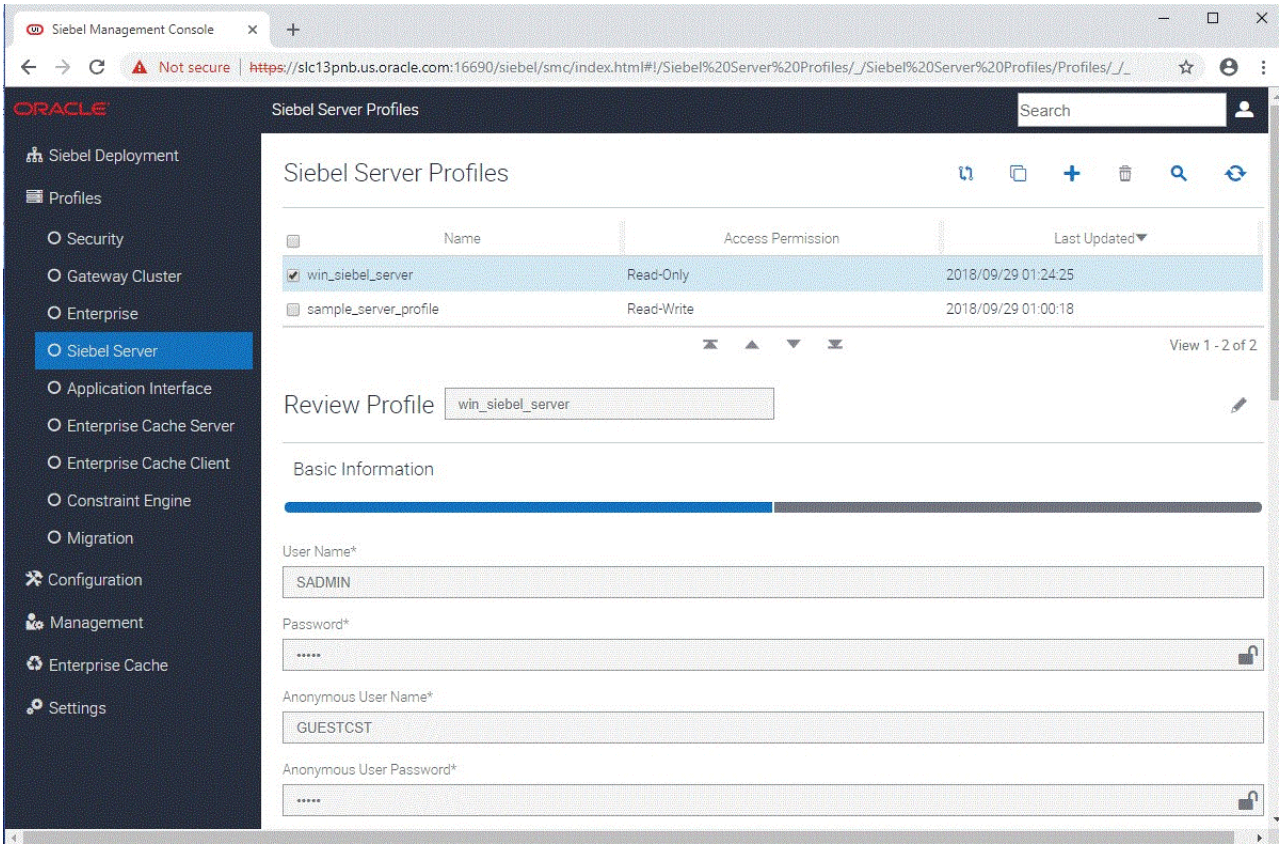
The following figure shows the Siebel Deployment screen in Siebel Management Console. Use this screen to deploy new items, using profiles that you have created, or to review or update settings for existing deployed items, where possible. When you select a deployment item that you have added, the icons change to include those that correspond to actions that you can perform on this deployment item.

A legend identifies the deployment status values that each deployment item might have, such as Staging, Deployment in Progress, Deployed, Deployment Failed, and so on. More information is provided later about some of these status values. The legend status values are shown when you point to the Legend label.



Profiles Screen

The following figure shows the Profiles screen in Siebel Management Console. Use this screen to create new profiles or to review or update settings for existing profiles. The navigation menu is shown in the side panel.



Configuration Screen

The figure in *About the Configuration Screen* shows the Configuration screen in Siebel Management Console. You can optionally use this screen to perform configuration activities for the Siebel Enterprise, Siebel Server, server components, and related areas. For more information, see *Configuring the Siebel Deployment*.

Management Screen

The figure in *About the Management Screen* shows the Management screen in Siebel Management Console. You can optionally use this screen to perform management activities for the Siebel Enterprise, Siebel Server, server components, and related areas. For more information, see *Managing the Siebel Deployment*.

Other Screens

Siebel Management Console also includes the Settings screen, which you can use to refresh the configuration and perform other tasks. For more information, see *Supported Activities in Siebel Management Console*.

Related Topics

About Configuring Siebel CRM

Starting the Siebel Management Console

Supported Activities in Siebel Management Console

Configuring the Siebel Deployment

Managing the Siebel Deployment

Accessibility for Siebel Management Console

Supported Activities in Siebel Management Console

Siebel Management Console supports several different activities or actions, most of which are summarized here, organized by the user interface features where these activities are initiated. This topic is part of *Running the Siebel Management Console*.

- Siebel Deployment selection in navigation menu (in the side panel), then overview area:
  - Reviewing overall deployment (overview area)
  - Adding a new deployment item (Add (+) icon in the overview area)
  - Deploying a profile into your Siebel environment (Deploy selection in Action field, then Submit button)
  - Staging a profile for later deployment into your Siebel environment (Stage selection in Action field, then Submit button)
  - Deleting configuration data for a deployed entity (Delete icon in the overview area). You might need to delete and redeploy a deployment item in order to modify its configuration in Siebel Management Console. Note that deleting a deployment item corresponding to a Siebel Server stops the system service.
  - Refreshing configuration data displayed in Siebel Management Console (Refresh icon in the overview area)
  - Reviewing the deployment status of items and nodes in your Siebel deployment (using the Legend link in the overview area)
- Profiles selection in navigation menu (in side panel), then overview area:
  - Comparing two or more profiles (Compare icon in the overview area)
  - Cloning a profile (Clone icon in the overview area)
  - Creating a profile for a configurable entity (Add icon (+) in the overview area)
  - Deleting a profile (Delete icon in the overview area)
  - Querying profile data (Query icon in the overview area)
  - Refreshing profile data (Refresh icon in the overview area). For example, you can use this option to retrieve the stored profile data in case another user had been editing profiles at the same time.
  - Editing a profile (Edit icon in the form view overview area)
- Configuration selection in navigation menu (in side panel). For more information, see *Configuring the Siebel Deployment*.
- Management selection in navigation menu (in side panel). For more information, see *Managing the Siebel Deployment*.
- Settings selection in navigation menu (in side panel), then overview area:
  - Invoking introspection of the Siebel deployment in order to refresh the deployment data in Siebel Management Console (Refresh Configuration button). You can use this option to bring deployment data into Siebel Management Console from configuration activities that had been done outside of Siebel Management Console, such as using Server Manager or the server administration views in the Siebel application.
  - Viewing the introspection log data (View Log button)
  - Removing the configuration for the Siebel Gateway (removes all profiles and deployment data)

- Modifying the search synchronization interval (in minutes, between 5 and 30), which specifies how often searchable configuration data is synchronized with the server. This setting affects searches for the current user and browser only.
- Specifying the Gateway Safe Mode user. Doing this enables login to Siebel Management Console in safe mode. For more information, see *Using Safe Mode for Siebel Management Console*.
- Updating the security profile for Siebel Gateway after logging in to Siebel Management Console as the safe mode user. For more information, see *Using Safe Mode for Siebel Management Console*.

- Toolbar area:
  - Searching the configuration data for available profiles or deployed entities
  - Viewing the current user of Siebel Management Console
  - Logging out of Siebel Management Console (Logout option for current user)

Related Topics

About Configuring Siebel CRM

Running the Siebel Management Console

Overview of Siebel Management Console

Configuring the Siebel Deployment

Managing the Siebel Deployment

Accessibility for Siebel Management Console

Using Safe Mode for Siebel Management Console

Safe mode for Siebel Management Console addresses the problem where an administrator cannot log in to Siebel Management Console because the back-end server for the authentication system, such as a database or an LDAP server, is unavailable. This task is part of *Running the Siebel Management Console*.

In this situation, you can start Siebel Management Console in safe mode and then update the Siebel Gateway security profile to specify the information necessary to connect to a working database or other authentication system. (Updating the Siebel Gateway security profile is the only action you can perform in safe mode.) Starting Siebel Management Console in safe mode uses a variation of the standard URL for starting Siebel Management Console. After performing the necessary tasks in safe mode, you can log in to Siebel Management Console normally to perform configuration or other tasks.

Logging in to Siebel Management Console in safe mode is not possible without first setting the Gateway Safe Mode user. For a new or existing deployment, you specify the safe mode user in the Settings screen of Siebel Management Console. Any administrator can become or create, modify, or delete a Gateway Safe Mode use. There is no default Gateway Safe Mode user.

> **Note:** After installing Siebel CRM, it is strongly recommended that you specify a safe mode user to help maintain access to Siebel Gateway as part of using Siebel Management Console.

- For more information about configuring the Siebel Management Console safe mode user and about using safe mode, see *Siebel Security Guide* .

- For more information about starting Siebel Management Console in safe mode, see *Starting the Siebel Management Console*.

Securing Siebel Management Console Deployments

It is recommended that you host SMC as a private, privileged administration endpoint on an Application Interface node in an internal administration tier.

For on-premises and cloud deployments, allow only authorized administrators to access SMC through an internal reverse proxy.

You must expose Application Interface nodes that participate in business transaction flows through public entry points only when appropriate network policies and security safeguards are in place.

Installing the Siebel Database

If you do not have an existing Siebel database, then you use the Siebel Database Configuration Wizard to install this database. This task is described in *Installing the Siebel Database on the RDBMS*.

Before you install the Siebel database, you must review the guidelines described for your RDBMS in *Configuring the RDBMS*.

This task is a step in *Roadmap: Installing Siebel CRM for a New Deployment*. When you are following the task sequence in this roadmap, depending on your authentication method, you might have to perform the Siebel database tasks before you can run the Siebel Management Console.

Related Topics

About Configuring Siebel CRM

About Installing the Siebel Database

About Database Updates for Siebel CRM

Starting the Siebel Database Configuration Wizard

Installing the Siebel Database on the RDBMS

Requirements for Installing and Configuring the Siebel Database

Configuring the RDBMS

Configuring the Siebel Gateway and Security

After you install Siebel CRM, you run the Siebel Management Console to configure the Siebel Gateway and the Security profile that it requires. You might also create additional Security profiles at some point after configuring the Siebel Gateway.

To configure the Siebel Gateway, perform the following tasks:

1. *Configuring the Siebel Gateway*
2. *Configuring a Security Profile*

> **Note:** In general, configuring the Siebel Gateway applies to new installations only. For a migration installation, the installer configures the Siebel Gateway and you do not need to configure the Siebel Gateway as described here. However, it is strongly recommended that you modify the existing Siebel Gateway configuration to specify the Gateway Safe Mode user. In addition, you might choose to create additional Security profiles.

For detailed information about some of the security settings in the Siebel Management Console, see *Siebel Security Guide* .

For more information about the activities that you can perform in the Siebel Management Console, including viewing configuration log data, see *Supported Activities in Siebel Management Console*.

This task is a step in:

- *Roadmap: Installing Siebel CRM for a New Deployment*
- *Roadmap: Installing Siebel CRM in an Upgrade Case (Existing Database Requiring Full Upgrade)*

Related Topics

About Configuring Siebel CRM

Running the Siebel Management Console

Configuring the Siebel Gateway Cluster

Requirements for Installing and Configuring Siebel Enterprise Components

Related Books

Siebel Security Guide

Configuring the Siebel Gateway

To configure the Siebel Gateway (which includes a Security profile), use the procedure that follows. This task is part of *Configuring the Siebel Gateway and Security*.

This task is required for new installations. It is not applicable for migration installations of Siebel Enterprise Server from a previous release (prior to Siebel CRM 17.0), because the migration installation task migrates the prior configuration. However, after migrating to the current release of Siebel CRM, it is strongly recommended that existing customers specify the safe mode user for Siebel Management Console, as described in *Siebel Security Guide* .

> **Note:** When you deploy Siebel Gateway clustering, then the Siebel Gateway that you configure here is referred to as the primary Siebel Gateway. This Siebel Gateway node participates in clustering, the same as other Siebel Gateway nodes. It is recommended that you configure the Siebel Gateway cluster after you configure the Siebel Gateway.

Related Topics

Running the Siebel Management Console

Using Safe Mode for Siebel Management Console

Configuring a Security Profile

Configuring the Siebel Gateway Cluster

Related Books

Siebel Security Guide

To configure the Siebel Gateway (first time running Siebel Management Console)

1. Run the Siebel Management Console, as described in *Starting the Siebel Management Console*. Log in using the basic authentication credentials that were provided during Siebel Application Interface installation.

   These credentials were specified in the Application Interface Authentication screen when you installed Siebel Application Interface, as described in *Installing Siebel CRM in a New Installation*.
2. Identify the Siebel Gateway host name and HTTPS redirect port number, in the format SiebelGatewayHost:Port, then click Save.
3. Create the security profile to configure access to the Siebel Gateway. Specify a name for the security profile.

   > **Note:** The security profile that is created on first login is named Gateway.

4. Click the plus sign, specify the data source information for the security profile, and then click Save. For more information about these settings, see *Configuration Settings for a Security Profile*.
   a. Specify the type of authentication to use. For example, if you have already created the Siebel database, then you could optionally select Database Authentication.
   b. Specify the name of the security adapter, such as DBSecAdpt. This name corresponds to a named subsystem. For more information, see *Siebel Security Guide* .
   c. Specify the data source (for database authentication).
   d. Specify whether to propagate changes (for database authentication).
   e. Specify the authorization roles.
   f. Test the user and password that you want to use under the specified authentication system.

   > **Note:** After saving these changes, you are forcibly logged out of the Siebel Management Console, in order to activate the security adapter. Go to *Running the Siebel Management Console* and log in again using the credentials you just specified. Then configure the Siebel Gateway registry, as identified in the next step.

5. Configure the Siebel Gateway registry, in which the configurations that you create in the Siebel Management Console will be stored, and then click Save:
   - Specify the Siebel Gateway registry port number. (If you configure Siebel Gateway clustering, then you can also specify this port number as part of configuring the Siebel Gateway cluster profile. See also *Configuring the Siebel Gateway Cluster*.)
   - Specify the user name that will access the registry.
   - Specify the password for this user.
   - Specify the primary language.
6. After configuring the Siebel Gateway and security, perform the configuration tasks for your Siebel deployment, which include the following:
   - Configure the safe mode user for Siebel Management Console. For more information, see *Siebel Security Guide* .

- Configure an additional Security profile. For more information, see *Configuring a Security Profile*.
- (Optional) Configure the Siebel Gateway cluster. For more information, see *Configuring the Siebel Gateway Cluster*.
- Configure the Siebel Enterprise. For more information, see *Configuring the Siebel Enterprise*.
- Configure the Siebel Server. For more information, see *Configuring the Siebel Server*.
- Configure the Siebel Application Interface. For more information, see *Configuring the Siebel Application Interface*.
- (Optional) Configure the Siebel Migration application. For more information, see *Configuring the Siebel Migration Application*.

Configuring a Security Profile

When you configure the Siebel Gateway, you configure the first Security profile. You can use the same Security profile for Siebel Enterprise authentication, or create another Security profile for the Siebel Enterprise. To create another Security profile, use the procedure that follows. This task is part of *Configuring the Siebel Gateway and Security*.

This task is required for new installations. It is not required for migration installations of Siebel Enterprise Server from a previous release, because the migration installation task migrates the prior configuration. However, you might choose to create a new Security profile.

Related Topics

Running the Siebel Management Console

Configuring the Siebel Gateway

To configure a Security profile

1. Run the Siebel Management Console, as described in *Starting the Siebel Management Console*.
2. Click Profiles in the navigation menu in the side panel, then click Security.

   Existing Security profiles are listed, such as the profile created for the deployed Siebel Gateway.
3. Click the plus (+) icon to add a new Security profile, or click the Clone button to clone an existing profile.
4. Specify a name for the profile.

   > **Note:** The security profile that is created on first login is named Gateway.

5. Click the plus (+) icon beside Data Sources to add a new data source.
6. Click Datasource.
7. Specify settings for the data source, as shown in the table in *Configuration Settings for a Security Profile*. Click Next when prompted.
8. When you have specified all applicable settings, click Submit to save your changes to the profile.

Configuration Settings for a Security Profile

The following table describes the settings that are requested in the Siebel Management Console for configuring a new Security profile or modifying an existing Security profile. Required fields display an asterisk (*) next to the field label. For

most of the options, pointing to the option displays tooltip text that provides a brief description of the field. This topic is part of *Configuring the Siebel Gateway and Security*.

Note: Database Authentication has two variants: Database Authentication Basic mode and Database Authentication Advanced mode. Unless otherwise specified, settings that are stated to apply to Database Authentication apply to both modes. For more information, see *Siebel Security Guide* .

Setting in Siebel Management Console	Section (Under Create Profile)	Comment or Description
Name	Data Sources > Data Source Name	Specify the name of the data source that will be created.
Type	Data Sources > Data Source Name	Specify the type of authentication you are using. The options are: • Database Authentication (Basic mode for development only) • Database Authentication (Advanced mode) • Lightweight Directory Access Protocol (LDAP) Authentication • Custom Security Authentication (using Security SDK)
Host Name	Data Sources > Data Source Name This option appears if you selected Database Authentication or LDAP Authentication.	Specify the host name for the data source, such as the host name of the database server for database authentication.
Port	Data Sources > Data Source Name This option appears if you selected Database Authentication or LDAP Authentication.	Specify the port number for the data source, such as the port number of the database server for database authentication.
SQL Style of Database	Data Sources > Data Source Name This option appears if you selected Database Authentication or Custom Authentication.	Specify the SQL style for your Siebel database. Specify one of the following: • Oracle Database Enterprise Edition • Microsoft SQL Server (select this for Azure SQL DB servers) • IBM DB2
Database Service Name	Data Sources > Data Source Name This option appears if you selected Database Authentication.	The service name of Oracle Enterprise Server edition, or the database name for IBM DB2 and Microsoft SQL Server or Azure SQL Database server.
Table Owner	Data Sources > Data Source Name	The table owner for the database.

Setting in Siebel Management Console	Section (Under Create Profile)	Comment or Description
	This option appears if you selected Database Authentication.	
Application User Distinguished Name (DN)	Data Sources > Data Source Name This option appears if you selected LDAP or Custom Authentication.	Specifies the user name of a record in the directory with sufficient permissions to read any user's information and do any necessary administration.
Application Password	Data Sources > Data Source Name This option appears if you selected LDAP or Custom Authentication.	Specifies the password for the user specified in the Application User Distinguished Name (DN) field.
Base Distinguished Name (DN)	Data Sources > Data Source Name This option appears if you selected LDAP or Custom Authentication.	Specifies the base distinguished name, which is the root of the tree under which users of this Siebel application are stored in the directory.
Custom Library	Data Sources > Data Source Name This option appears if you selected Custom Authentication.	Name of the custom security adapter implementation.
CRC Checksum	Data Sources > Data Source Name	Provide the value of the checksum performed on the applicable security adapter library (DLL). This value, applicable for the Siebel Server only, ensures that each user accesses the Siebel database through the correct security adapter. If this field is empty or contains the value 0 (zero), then no checksum validation is performed. If you upgrade your version of Siebel CRM, then you must recalculate the checksum value and replace the value in this field.
Credentials Attribute	Data Sources > Data Source Name This option appears if you selected LDAP or Custom Authentication.	Specifies the attribute type that stores a database account.
Hash DB Password	Data Sources > Data Source Name This option appears if you selected LDAP or Custom Authentication.	Specifies password hashing for database credentials passwords.
Hash User Password	Data Sources > Data Source Name This option appears if you selected Database Authentication, LDAP, or Custom Authentication.	Specifies password hashing for user passwords.

Setting in Siebel Management Console	Section (Under Create Profile)	Comment or Description
Hash Algorithm	Data Sources > Data Source Name This option appears if you selected Hash DB Password or Hash User Password.	Specifies the hash algorithm to be used for password hashing. Choose one of the following: • SHA2 • SHA1
Password Attribute Type	Data Sources > Data Source Name This option appears if you selected LDAP or Custom Authentication.	Specifies the attribute type that stores a database account.
Propagate Change	Data Sources > Data Source Name This option appears if you selected LDAP or Custom Authentication.	Specifies whether to allow administration of the directory through the Siebel application user interface.
Roles Attribute (optional)	Data Sources > Data Source Name This option appears if you selected LDAP or Custom Authentication.	Specifies the attribute type for roles stored in the directory.
Shared Database Account Distinguished Name (fully qualified domain name)	Data Sources > Data Source Name This option appears if you selected LDAP or Custom Authentication.	Specifies the absolute path of an object in the directory that has the shared database account for the application.
Shared DB User Name	Data Sources > Data Source Name This option appears if you selected Configure Web Single-Sign-On for Database Authentication Advanced, LDAP, or Custom Authentication.	Specifies the user name to connect to the Siebel database.
Shared DB Password	Data Sources > Data Source Name This option appears if you selected Configure Web Single-Sign-On for Database Authentication Advanced, LDAP, or Custom Authentication.	Specifies the password for the shared database user name.
Security Adapter Mapped User Name	Data Sources > Data Source Name This option appears if you selected LDAP or Custom Authentication.	Specifies that, when the user key passed to the security adapter is not the Siebel user name, the security adapter retrieves the user name for authenticated users from an attribute defined by the parameter SiebelUsernameAttributeType.
Siebel User Name Attribute	Data Sources > Data Source Name This option appears if you selected Security Adapter Mapped User Name for LDAP or Custom Authentication.	Specifies the attribute from which the security adapter retrieves an authenticated user's Siebel user ID.

Setting in Siebel Management Console	Section (Under Create Profile)	Comment or Description
SSL	Data Sources > Data Source Name This option appears if you selected LDAP Authentication.	Specifies to enable Secure Sockets Layer for socket connections to the host.
Wallet Password	Data Sources > Data Source Name This option appears if you selected SSL for LDAP or Custom Authentication.	Specifies the password to open the wallet that contains a certificate for the certification authority used by the directory server.
Configure Web Single Sign-On (Web SSO)	Data Sources > Data Source Name This option appears if you selected Database Authentication Advanced, LDAP, or Custom Authentication.	Specifies that the security adapter uses Web Single Sign-On (Web SSO) authentication rather than security adapter authentication.
Trust Token	Data Sources > Data Source Name This option appears if you selected Configure Web Single-Sign-On for Database Authentication Advanced, LDAP, or Custom Authentication.	Specifies a password to be used with Web Single Sign-On (Web SSO) authentication.
Salt Attribute Type	Data Sources > Data Source Name This option appears if you selected LDAP or Custom Authentication.	Specifies the attribute that stores the salt value if you are using password salting.
Salt User Password	Data Sources > Data Source Name This option appears if you selected LDAP or Custom Authentication.	Specifies whether to add salt values to user passwords before they are hashed.
User Name Attribute Type	Data Sources > Data Source Name This option appears if you selected LDAP or Custom Authentication.	Specifies the attribute type under which the user's login name is stored in the directory.
Enterprise Security Authentication Profile	Basic Information	Select one of the following options for your enterprise security authentication: • Database Authentication (Basic mode for development only) • Database Authentication (Advanced mode) • Lightweight Directory Access Protocol (LDAP) Authentication • Custom Security Authentication (using Security SDK) **Note:** You must create the security profile before you configure the Siebel Enterprise.

Setting in Siebel Management Console	Section (Under Create Profile)	Comment or Description
Security Adapter Name (named subsystem)	Basic Information	Select the security adapter name. • For Database Authentication Basic and Advanced modes: DBSecAdpt • For LDAP: LDAPSecAdpt • For Custom Authentication: CustSecAdpt
Database Security Adapter Data Source	Basic Information This option appears if you selected Database Authentication.	Select the security adapter data source.
Database Security Adapter Propagate Changes	Basic Information This option appears if you selected Database Authentication.	Specify whether to propagate changes for the security adapter.
Authorization Roles (comma-separated)	Basic Information	Specify one or more authorization roles for accessing the Siebel Gateway. Default value: Siebel Administrator (which is provided by Oracle and cannot be changed). For more information, see *Requirements for Siebel Gateway Authentication*.
User Name	Testing	Specify the user name for testing authentication under the specified authentication system.
Password	Testing	Specify the password for the user account used for testing.
Connection String	Data Sources > Data Source Name This option appears if you selected Database Authentication Advanced.	Specify the connection string for the Siebel database. For more information, see the information about configuring Web Single Sign-On with database authentication in *Siebel Security Guide* .

CAUTION: If the Siebel Gateway security profile is configured to use Database Authentication Advanced mode and you need to restore a release prior to Siebel CRM 20.10, then you must modify the Siebel Gateway security profile to specify a different authentication option (such as Database Authentication Basic mode). If you do not do so, then the Gateway service will fail. You can modify the Siebel Gateway security profile to specify the other authentication system before or after restoring the prior release. To do so after restoring the prior release, you must use safe mode for Siebel Management Console, as described in *Using Safe Mode for Siebel Management Console*. You must perform this task before you perform the migration installation of Siebel Application Interface for the existing installation of SWSE.

Configuring the Siebel Enterprise

The task for configuring the Siebel Enterprise lets you configure authentication using Siebel security adapters, TLS encryption, and related settings.

To configure the Siebel Enterprise, perform the following tasks:

1. *Creating a Siebel Enterprise Profile*
2. *Deploying the Siebel Enterprise*

Note: Configuring the Siebel Enterprise applies to new installations only. For a migration installation, the installer migrates the Siebel Enterprise configuration.

For detailed information about the security settings in the Siebel Management Console, see *Siebel Security Guide* .

For more information about the activities that you can perform in the Siebel Management Console, including viewing configuration log data, see *Supported Activities in Siebel Management Console*.

This task is a step in:

- *Roadmap: Installing Siebel CRM for a New Deployment*
- *Roadmap: Installing Siebel CRM in an Upgrade Case (Existing Database Requiring Full Upgrade)*

Related Topics

About Configuring Siebel CRM

Running the Siebel Management Console

Requirements for Installing and Configuring Siebel Enterprise Components

Related Books

Siebel Security Guide

Siebel System Administration Guide

Creating a Siebel Enterprise Profile

This procedure describes how to configure a Siebel Enterprise profile. You can create multiple profiles, but you can deploy only one instance of Siebel Enterprise in your Siebel deployment. This task is part of *Configuring the Siebel Enterprise*.

Related Topics

Running the Siebel Management Console

Deploying the Siebel Enterprise

To create a Siebel Enterprise profile

1. Run the Siebel Management Console, as described in *Starting the Siebel Management Console*.

2. Click Profiles in the navigation menu in the side panel, then click Enterprise.
   Existing Enterprise profiles are listed, if any.
3. Click the plus (+) icon to add a new Enterprise profile, or click the Clone button to clone an existing profile.
4. Specify a name for the profile.
5. Specify additional settings, as shown in the table in *Configuration Settings for a Siebel Enterprise Profile*.
6. When you have specified all applicable settings, click Submit to save your changes to the profile.

Configuration Settings for a Siebel Enterprise Profile

The following table describes the settings that are requested in the Siebel Management Console for configuring a new Siebel Enterprise profile or modifying an existing Siebel Enterprise profile. Required fields display an asterisk (*) next to the field label. For most of the options, pointing to the option displays tooltip text that provides a brief description of the field. This topic is part of *Configuring the Siebel Enterprise*.

For requirements and restrictions for some of these parameters, see *Requirements for Installing and Configuring Siebel Enterprise Components*. Parameters that are set as a result of configuring the Siebel Enterprise are set at the Siebel Enterprise level or set for named subsystems such as for a data source or security adapter.

Note: After you have already configured the Siebel Enterprise, you can modify its configuration using Siebel Server Manager, as described in *Siebel System Administration Guide* .

Setting in Siebel Management Console	Section (Under Create Profile)	Comment or Description
Primary Siebel File System	Basic Information	Specify the location or locations of the Siebel File System. If the folder path provided does not exist, then the deployment step creates it. For more information, see *Creating the Siebel File System*.
User Name	Authentication	Specify the authorized user name for logging into the Siebel Gateway.
Password	Authentication	Specify the password for this user.
RDBMS Platform	Database Information	Specify the RDBMS platform for the Siebel database. The options are: • Oracle Database Enterprise Edition • Microsoft SQL Server (use this for Azure SQL Database Server) • IBM DB2 UDB for Linux UNIX Windows • IBM DB2 UDB for z/OS
Oracle SQLNet Connect String	Database Information	(Oracle Database only) Specify the Oracle SQLNet connect string.
Microsoft SQL Server Host Name	Database Information	(Microsoft SQL and Azure SQL Database Servers only)

Setting in Siebel Management Console	Section (Under Create Profile)	Comment or Description
		Specify the host name for Microsoft SQL Server.
Microsoft SQL Server Port	Database Information	(Microsoft SQL and Azure SQL Database Servers only) Specify the database port number for Microsoft SQL Server.
Microsoft SQL Server Database Name	Database Information	(Microsoft SQL and Azure SQL Database Servers only) Specify the database name for Microsoft SQL Server.
IBM DB2 Database Alias	Database Information	(IBM DB2 only) Specify the database alias for IBM DB2.
Current IBM DB2 SQL ID / Group Name	Database Information	(IBM DB2 for z/OS only) Specify the current SQL ID or group name for IBM DB2.
IBM DB2 Client Instance Home Directory	Database Information	(IBM DB2 or DB2 for z/OS only) Provide the absolute path of the parent directory where IBM DB2 client software is installed on UNIX, or the value of the DB2INSTPROF parameter on Windows. UNIX example: **`/export/home/myuser`** Windows example: **`C:\Program Data\IBM\DB2\Copy Name`**
Database User Account Name	Database Information	(For all database platforms) Specify the database user account.
Database User Account Password	Database Information	(For all database platforms) Specify the password for the database user account.
Table Owner	Database Information	(Oracle Database, DB2, or DB2 for z/OS only) Specify the table owner.
Authentication Profile	Authentication	Specify the security profile to use for Siebel Enterprise authentication.
Primary Language	Authentication	Specify the primary language for your Siebel deployment.
Security Encryption Level or Type	Security Information	Specify the security encryption level or type. Select one of the following options:

Setting in Siebel Management Console	Section (Under Create Profile)	Comment or Description
		• SISNAPI Without Encryption • SISNAPI Using TLS 1.2 (default)
Certification Authority (CA) Certificate File Name	Security Information This option and all of the remaining items appear only if you selected the TLS option in Security Encryption Level or Type.	Provide the name that can identify the trusted authority who issued the certificate file. For more information, see *Siebel Security Guide* .
Private Key File Name	Security Information	Provide the name of the PEM private key file. For more information, see *Siebel Security Guide* .
Private Key File Password	Security Information	Provide the password that can decrypt the PEM private key file. For more information, see *Siebel Security Guide* .
Enable Peer Authentication	Security Information	Check this field to enable peer authentication during the SSL handshake. For more information, see *Siebel Security Guide* .
Validate Peer Certificate	Security Information	Check this field to verify that the host name matches with the host name information presented in the certificate. For more information, see *Siebel Security Guide* .

Deploying the Siebel Enterprise

This procedure describes the steps for deploying Siebel Enterprise. You can deploy only one instance of Siebel Enterprise in your Siebel deployment. This task is part of *Configuring the Siebel Enterprise*.

This task is required for new installations. It is not applicable for migration installations of Siebel Enterprise Server from a previous release, because the migration installation task migrates the prior configuration.

Before you can deploy the Siebel Enterprise, you must have already configured the Siebel Gateway, as described in *Configuring the Siebel Gateway and Security*.

> **Note:** After deployment, the Siebel Enterprise profile is in a read-only state. If you need to update the configuration, then it is recommended to use Server Manager. Then refresh the configuration data displayed in Siebel Management Console.

Related Topics

Additional Tasks for Migration Installations

About Configuring Siebel CRM

Running the Siebel Management Console

Creating a Siebel Enterprise Profile

To deploy the Siebel Enterprise

1. Run the Siebel Management Console, as described in *Starting the Siebel Management Console*.
2. Click Siebel Deployment in the navigation menu.
3. In the overview area, click the plus sign (+) and then select Enterprise to add a new instance to your Siebel deployment.
4. For Action, specify whether you will deploy this Siebel Enterprise or stage it for later deployment:
   - If you want to stage this Siebel Enterprise for later deployment, then click Staging.
   - If you want to deploy this Siebel Enterprise to your Siebel deployment now, then click Deploy.
5. For Profile, select the profile to use for this instance of Siebel Enterprise.
   You created this profile in *Creating a Siebel Enterprise Profile*.
6. For Siebel Enterprise Name, specify the name of this Siebel Enterprise.
   For more information, see *Restrictions on Names for Siebel Enterprise and Siebel Server*.
7. For Enterprise Description, specify an optional description of this Siebel Enterprise.
8. To continue, click Submit. Otherwise, click Cancel.
   After the configuration has been submitted, the state of this Siebel Enterprise deployment item changes to one of the following states, based on your selection in Step 4: Staging or Deployment in Progress.
   - If the deployment completes successfully, then, when you refresh the configuration, the state changes to Deployed.
   - If the deployment fails, then the state changes to Deployment Failed.
   - For a deployment in the Staging state, you can deploy it by either clicking the Deploy icon or by setting the Action to Deploy and resubmitting the deployment.
9. (Optional) View log data about this Siebel Enterprise deployment.

Configuring the Siebel Server

After you have installed Siebel Server and performed the prerequisite tasks, you run the Siebel Management Console and choose tasks to configure the Siebel Server, as described in this topic. To configure the Siebel Server, perform the following tasks:

1. *Creating a Siebel Server Profile*
2. *Deploying the Siebel Server*

> **Note:** Configuring the Siebel Server applies to new installations. For a migration installation, the installer migrates the Siebel Server configuration.

For detailed information about the security settings in the Siebel Management Console, see *Siebel Security Guide* .

For more information about the activities that you can perform in the Siebel Management Console, including viewing configuration log data, see *Supported Activities in Siebel Management Console*.

This task is a step in:

- *Roadmap: Installing Siebel CRM for a New Deployment*

- *Roadmap: Installing Siebel CRM in an Upgrade Case (Existing Database Requiring Full Upgrade)*

Related Topics

About Configuring Siebel CRM

Running the Siebel Management Console

Requirements for Installing and Configuring Siebel Enterprise Components

Related Books

Siebel Security Guide

Siebel System Administration Guide

Creating a Siebel Server Profile

This procedure describes how to configure a Siebel Server profile. The same steps apply whether you are configuring the first Siebel Server profile or configuring any additional profile. This task is part of *Configuring the Siebel Server*.

Related Topics

Running the Siebel Management Console

Deploying the Siebel Server

To create a Siebel Server profile

1. Run the Siebel Management Console, as described in *Starting the Siebel Management Console*.
2. Click Profiles in the navigation menu in the side panel, then click Siebel Server.
   Existing Siebel Server profiles are listed, if any.
3. Click the plus (+) icon to add a new Siebel Server profile, or click the Clone button to clone an existing profile.
4. Specify a name for the profile.
5. Specify additional settings, as shown in *Configuration Settings for a Siebel Server Profile*.
6. When you have specified all applicable settings, click Submit to save your changes to the profile.

Configuration Settings for a Siebel Server Profile

The following table describes the settings that are requested in the Siebel Management Console for configuring a new Siebel Server profile or modifying an existing Siebel Server profile. Required fields display an asterisk (*) next to the field label. For most of the options, pointing to the option displays tooltip text that provides a brief description of the field. This topic is part of *Configuring the Siebel Server*.

For requirements for and restrictions on some of these parameters, see *Requirements for Installing and Configuring Siebel Enterprise Components*. Parameters that are set as a result of configuring the Siebel Server are set at the Siebel Server level.

> **Note:** After you have already configured a Siebel Server, you can modify its configuration using Siebel Server Manager, as described in *Siebel System Administration Guide* .

Setting in Siebel Management Console	Section (Under Create Profile)	Comment or Description
User Name	Basic Information	Specify the authorized user name for logging into the Siebel Gateway.
Password	Basic Information	Specify the password for this user.
Anonymous User Name	Basic Information	Enter the user name required for anonymous browsing and initial access to the login pages of the application you are implementing. For more information about anonymous users, see *Requirements for Configuring Anonymous Users for the Siebel Application Interface*.
Anonymous User Password	Basic Information	Specify the password for this anonymous user.
Enable Component Groups	Basic Information	Specify one or more component groups to enable for the Siebel Server. For more information about component groups, see *Siebel System Administration Guide* .
Siebel Connection Broker Port	Basic Information	Specify the TCP/IP port number for the Siebel Connection Broker component (alias SCBroker). The default port number is 2321. This port number is used for all of the communications between SCBroker and other components, including the Siebel Application Interface. Make sure that the port number that you specify is not already used by any other applications or components. You also specify this port number when you configure the Siebel Application Interface, as described in *Deploying the Siebel Application Interface*. For more information about the SCBroker component, see *Siebel Deployment Planning Guide* and *Siebel System Administration Guide* .
Network TCP/IP Port for Synchronization Manager	Basic Information	Specify the TCP/IP port number for the Synchronization Manager component, which is part of Siebel Remote. Make sure that the port number that you specify is not already used by any other applications or components. The default port number is 40400.
Server-Specific Security Encryption Settings	Enhanced Settings > Security	Select this option to configure security and encryption for communications between this Siebel Server and other servers. If you do not select this option, then the applicable settings are inherited from the Enterprise.
Server-Specific Security Authentication Profile Assignment	Enhanced Settings > Security	Select this option to assign an existing security adapter to this Siebel Server or to specific components.
Security Encryption Level or Type	Enhanced Settings > Security	Specify the security encryption level or type. Select one of the following options:

Setting in Siebel Management Console	Section (Under Create Profile)	Comment or Description
	This option appears if you selected Server-Specific Security Encryption Settings.	• SISNAPI Without Encryption • SISNAPI Using TLS 1.2 (default) For information about configuring TLS for Siebel Remote, see *Siebel Remote and Replication Manager Administration Guide* and *Siebel Security Guide* .
Certificate File Name	Enhanced Settings > Security This option appears if you selected the TLS option in Security Encryption Level or Type.	Provide the name of the ASN/PEM certificate file. For more information, see *Siebel Security Guide* .
Certification Authority (CA) Certificate File Name	Enhanced Settings > Security This option appears if you selected the TLS option in Security Encryption Level or Type.	Provide the name that can identify the trusted authority who issued the certificate file. For more information, see *Siebel Security Guide* .
Clustering Configuration	Enhanced Settings > Clustering	Specify the clustering configuration. You can select one of the following clustering options: • Not Clustered • Clustered Using Virtual Host Name (valid NETBIOS name) • Clustered Using Virtual IP Address (for example, 192.168.0.1) In certain cases, it is useful to cluster Siebel Servers for high availability. The cluster nodes must be started per cluster instructions, if clustering is selected. For more information about clustering Siebel Servers, see *Siebel Deployment Planning Guide* .
Siebel Server Cluster Virtual Host Name/IP	Enhanced Settings > Clustering This option appears if you selected either of the two clustering options.	Specify one of the following, depending on your selection under Clustering Configuration: • If you selected Clustered Using Virtual Host Name, then specify the host name. • If you selected Clustered Using Virtual IP Address, then specify the IP address.
RDBMS Platform	Enhanced Settings > Other Information	Specify the RDBMS platform for the Siebel database. The options are: • Oracle Database Enterprise Edition • Microsoft SQL Server (Use this for Azure SQL Database servers) • IBM DB2

Setting in Siebel Management Console	Section (Under Create Profile)	Comment or Description
Register External Oracle ODBC Driver	Enhanced Settings > Other Information This option appears if you selected Oracle Database Enterprise Edition.	This option is read-only and is set to True.
Microsoft SQL Server Port Number	Enhanced Settings > Other Information This option appears if you selected Microsoft SQL Server.	Provide the port number for Microsoft SQL Server or Azure SQL Database servers.
IBM DB2 Client Instance Home Directory	Enhanced Settings > Other Information This option appears if you selected IBM DB2.	Provide the absolute path of the parent folder where the IBM DB2 client is installed. This setting is optional for Microsoft Windows.

Deploying the Siebel Server

This procedure describes the steps for deploying an installed instance of Siebel Server. You perform the same procedure for each installed instance of Siebel Server. This task is part of *Configuring the Siebel Server*.

This task is required for new installations. It is not applicable for migration installations of Siebel Enterprise Server from a previous release, because the migration installation task migrates the prior configuration. However, you might choose to install and deploy a new Siebel Server.

Before you can deploy an installed instance of Siebel Server, you must have already configured the Siebel Enterprise, as described in *Configuring the Siebel Enterprise*.

> **Note:** After deployment, the Siebel Server profile is in a read-only state. If you need to update the configuration, then it is recommended to use Server Manager. Then refresh the configuration data displayed in Siebel Management Console.

Related Topics

Additional Tasks for Migration Installations

About Configuring Siebel CRM

Running the Siebel Management Console

Creating a Siebel Server Profile

To deploy the Siebel Server

1. Run the Siebel Management Console, as described in *Starting the Siebel Management Console*.

2. Click Siebel Deployment in the navigation menu.
3. In the overview area, click the plus sign (+) and then select Siebel Server to add a new instance to your Siebel deployment.

   The Siebel Server must already have been installed.
4. For Host Name:HTTPS Port, specify the host name and HTTPS redirect port for this instance of Siebel Server. You specified this HTTPS redirect port during Siebel Server installation.
5. For Profile, select the profile to use for this deployment of Siebel Server.

   You created this profile in *Creating a Siebel Server Profile*.
6. For Action, specify whether you will deploy this Siebel Server or stage it for later deployment:
   - If you want to stage this Siebel Server for later deployment, then click Staging.
   - If you want to deploy this Siebel Server to your Siebel deployment, then click Deploy.
7. For Siebel Server Name, specify the name of this Siebel Server.

   For more information, see *Restrictions on Names for Siebel Enterprise and Siebel Server*.
8. For Siebel Server Description, specify an optional description of this Siebel Server.
9. For Deployed Languages, specify all of the languages that you plan to deploy for this Siebel Server.
10. To continue, click Submit. Otherwise, click Cancel.

    After the configuration has been submitted, the state of this Siebel Server deployment item changes to one of the following states, based on your selection in Step 6: Staging or Deployment in Progress.
    - If the deployment completes successfully, then, when you refresh the configuration, the state changes to Deployed. The Siebel Server system service starts automatically.
    - If the deployment fails, then the state changes to Deployment Failed.
    - For a deployment in the Staging state, you can deploy it by either clicking the Deploy icon or by setting the Action to Deploy and resubmitting the deployment.
11. (Optional) View log data about this Siebel Server deployment.

Configuring the Siebel Application Interface

After you have installed Siebel Application Interface, and after you have configured the Siebel Gateway, Siebel Enterprise, and Siebel Server, you run the Siebel Management Console and choose tasks to configure the Siebel Application Interface, as described in this topic. To configure the Siebel Application Interface, perform the following tasks:

1. *Creating a Siebel Application Interface Profile*
2. *Deploying the Siebel Application Interface*

For information about deploying multiple instances of Siebel Application Interface, see *Planning the Siebel Application Interface Topology*.

For more information about the activities that you can perform in the Siebel Management Console, including viewing configuration log data, see *Supported Activities in Siebel Management Console*.

This task is a step in:

- *Roadmap: Installing Siebel CRM for a New Deployment*

- *Roadmap: Installing Siebel CRM in a Migration Installation Case (Existing Installation of Siebel CRM 16.x or Earlier)*
- *Roadmap: Installing Siebel CRM in an Upgrade Case (Existing Database Requiring Full Upgrade)*

Related Topics

Additional Tasks for Migration Installations

About Configuring Siebel CRM

Running the Siebel Management Console

Customizing the Application Container for Siebel Application Interface

Requirements for Installing and Configuring the Siebel Application Interface

Planning the Siebel Application Interface Topology

Related Books

Siebel REST API Guide

Siebel Security Guide

Siebel System Administration Guide

Siebel System Monitoring and Diagnostics Guide

Creating a Siebel Application Interface Profile

This procedure describes how to configure a Siebel Application Interface profile. The same steps apply whether you are configuring the first Siebel Application Interface profile, reconfiguring the same profile with different settings, or configuring any additional profile. This task is part of *Configuring the Siebel Application Interface*.

> **Note:** Before you create a Siebel Application Interface profile, run the Server Manager and make sure that the Siebel Server has been deployed and is running, along with its Application Object Manager server components. If these components are running, then the Object Manager and Application settings in Siebel Application Interface profile configuration are populated with values you can choose from that reflect the available components on the Siebel Server. For information about using Server Manager, see *Siebel System Administration Guide* .

Related Topics

Running the Siebel Management Console

Deploying the Siebel Application Interface

To create a Siebel Application Interface profile

1. Run the Siebel Management Console, as described in *Starting the Siebel Management Console*.
2. Click Profiles in the navigation menu in the side panel, then click Application Interface.

   Existing Siebel Application Interface profiles are listed, if any.
3. Click the plus (+) icon to add a new Siebel Application Interface profile, or click the Clone button to clone an existing profile.

4. Specify a name for the profile.
5. Specify additional settings, as shown in the table in *Configuration Settings for a Siebel Application Interface Profile*.
6. When you have specified all applicable settings, click Submit to save your changes to the profile.

Configuration Settings for a Siebel Application Interface Profile

The following table describes the settings that are requested in the Siebel Management Console for configuring a new Siebel Application Interface profile or modifying an existing Siebel Application Interface. This topic is part of *Configuring the Siebel Application Interface*.

Required fields display an asterisk (*) next to the field label. For most of the options, pointing to the option displays tooltip text that provides a brief description of the field. Clicking the large plus (+) icon next to some options adds a new instance of a configurable item, which is represented in the second column using italics.

Default values have changed for some of these settings in recent releases, as follows:

- The default setting changed for HTTP 1.1-Compliant Firewall / Enable Web Compression to be enabled by default. You can optionally update any Siebel Application Interface profiles created before Siebel CRM 21.3 to enable this setting.
- The default values changed for Method to Check Server Availability and Command to Check Server Availability to the values shown in this table. As a consequence, customers must update any Siebel Application Interface profiles created before Siebel CRM 20.2 to reflect the values shown in the table. Failing to do so could result in idle sessions not getting logged out in timely fashion.

Setting in Siebel Management Console	Section (Under Create Profile)	Comment or Description
HTTP 1.1-Compliant Firewall / Enable Web Compression	Basic Information	Specify that the Siebel Application Interface compresses HTTP traffic. Compressing HTTP traffic, where it is feasible to do so, substantially reduces bandwidth consumption. This feature is supported on HTTP 1.1 and is not supported on HTTP 1.0. Default value: enabled. **Note:** This feature compresses only dynamic HTML content that is served by the application container. To compress static content like cascading style sheets (CSS files) and Javascript (JS files), you must change the Tomcat connector configuration in server.xml of the application container for Siebel Application Interface. For details, see *Compressing Static Content for Siebel Application Interface*. For more information about specifying compression, see *Siebel Security Guide* .
Configure Fully Qualified Domain Name	Basic Information	Specify whether you are using a fully qualified domain name (FQDN) for your Siebel CRM deployment. It is strongly recommended to use the FQDN feature. For more information, see *Configuring Siebel CRM for Pop-Up Blocker Compatibility*.

Setting in Siebel Management Console	Section (Under Create Profile)	Comment or Description
Fully Qualified Domain Name	Basic Information This option appears if you selected Configure Fully Qualified Domain Name.	Specify the fully qualified domain name (FQDN) for your Siebel CRM deployment, such as example.com. It is strongly recommended to use the FQDN feature.
Active Session Timeout Value	Authentication	Specify the total number of seconds that a session can remain inactive before the user is logged out and the session is closed. Default value: 900 (in seconds)
Login Session (guest session) Timeout Value	Authentication	Specify the timeout value for the login session. Default value: 300 (seconds)
Method to Check Server Availability	Authentication	Specify the method to use for checking server availability. Default value: UpdatePrefMsg
Command to Check Server Availability	Authentication	Specify the command to use for checking server availability. Default value: HeartBeat
Session Token Usage Duration	Authentication	Specify the usage duration for the session token. Default value: 2880 (minutes)
Session Token Timeout Value	Authentication	Specify the timeout value for the session token. Default value: 900 (seconds)
Maximum Possible Tabbed Sessions	Authentication	Specify the maximum possible number of tabs for multiple tab browsing. Default value: 1 This setting is effective only when the EnableMultiTab server parameter is set to True for the specified Application Object Manager. For more information about configuring multiple tab browsing, see *Configuring Siebel Open UI* .
Configure Web Single Sign-On (Web SSO)	Authentication	Specify whether to use Web single sign-on.
Anonymous User Name	Authentication	Specify the Siebel user ID that starts the anonymous session from which an end user is shown the login page for an application. For example: GUESTCST Later in profile configuration, you can optionally configure a different anonymous user for individual applications, if you require settings different from the defaults. For example, you might use a different anonymous user for customer applications than for employee applications. For more information about anonymous users, see *Requirements for Configuring Anonymous Users for the Siebel Application Interface*.

Setting in Siebel Management Console	Section (Under Create Profile)	Comment or Description
Anonymous User Password	Authentication	Specify the password for the anonymous user.
Trust Token	Authentication This option appears if you selected Configure Web Single Sign-On.	Specify the trust token to use for Web SSO authentication.
User Specification	Authentication This option appears if you selected Configure Web Single Sign-On.	Specify the user specification to use for Web SSO authentication.
Anonymous User Name	Authentication > REST Inbound Authentication	Specify the anonymous user to use for REST inbound authentication. For more information about anonymous users, see *Requirements for Configuring Anonymous Users for the Siebel Application Interface*. See also *Siebel REST API Guide* .
Anonymous User Password	Authentication > REST Inbound Authentication	Specify the password for the anonymous user for REST inbound authentication.
Authentication Type	Authentication > REST Inbound Authentication	Specify the authentication type for REST inbound authentication. You can select one of the following options: • Basic Authentication • Single Sign-On • OAuth For more information, see *Siebel REST API Guide* .
Introspection URL	Authentication > REST Inbound Authentication This option appears if you selected OAuth.	Specify the introspection URL for REST inbound authentication. For more information, see *Siebel REST API Guide* .
Client Id	Authentication > REST Inbound Authentication This option appears if you selected OAuth.	Specify the client ID, which is created when Siebel CRM is registered as a confidential client in the OAuth server. For more information, see *Siebel REST API Guide* .
Client Secret	Authentication > REST Inbound Authentication This option appears if you selected OAuth.	Specify the client secret, which is created when Siebel CRM is registered as a confidential client in the OAuth server. For more information, see *Siebel REST API Guide* .

Setting in Siebel Management Console	Section (Under Create Profile)	Comment or Description
Trust Token	Authentication > REST Inbound Authentication This option appears if you selected Single Sign-On or OAuth.	Specify the trust token to use for REST inbound authentication. Use the same value specified for the security adapter. For more information, see *Siebel REST API Guide* .
User Specification	Authentication > REST Inbound Authentication This option appears if you selected Single Sign-On.	Specify the user specification to use for REST inbound authentication. For more information, see *Siebel REST API Guide* .
Session Timeout (seconds)	Authentication > REST Inbound Authentication	Specify the Object Manager session timeout, in seconds, to use for REST inbound authentication. For more information, see *Siebel REST API Guide* .
Secure Channel	Authentication > REST Inbound Authentication	Specify whether to use a secure channel. Enforces OAuth server certificate validation, as required for introspection. For more information, see *Siebel REST API Guide* .
User Interface Log Level	Logging	Specify the user interface logging level. Select one of the following options: Fatal, Error (default), Warning, Debug, Trace, Information, All
EAI Log Level	Logging	Specify the EAI logging level. Select one of the following options: Fatal, Error, Warning, Debug, Trace, Information, All
DAV Log Level	Logging	Specify the DAV logging level. Select one of the following options: Fatal, Error (default), Warning, Debug, Trace, Information, All
REST Outbound Log Level	Logging	Specify the REST outbound logging level. Select one of the following options: Fatal, Error (default), Warning, Debug, Trace, Information, All
REST Inbound Log Level	Logging	Specify the REST inbound logging level. Select one of the following options: Fatal, Error (default), Warning, Debug, Trace, Information, All
SOAP Outbound Log Level	Logging	Specify the SOAP outbound logging level. Select one of the following options: Fatal, Error (default), Warning, Debug, Trace, Information, All
Object Manager	REST Inbound Defaults	Select the Object Manager component to use for REST inbound communications, such as EAI Object Manager. For more information, see *Siebel REST API Guide* .

Setting in Siebel Management Console	Section (Under Create Profile)	Comment or Description
REST Response Base URL	REST Inbound Defaults	Specify the base URL for a REST response. For more information, see *Siebel REST API Guide* .
Maximum Possible Connections	REST Inbound Defaults	Specify the REST connection pool size. Default value: 20. For more information, see *Siebel REST API Guide* .
Method Name	REST Inbound Defaults > REST Resource Parameter List > Query	Specify the method name to use for queries. For more information, see *Siebel REST API Guide* .
Name	REST Inbound Defaults > REST Resource Parameter List > Query > Parameter List	Specify the name for each query parameter. For more information, see *Siebel REST API Guide* .
Alias	REST Inbound Defaults > REST Resource Parameter List > Query > Parameter List	Specify the alias for each query parameter. For more information, see *Siebel REST API Guide* .
Application Name	Applications > Language > Basic Information	Specify an application to configure for this instance of Siebel Application Interface, for each specified language.
Object Manager	Applications > Language > Basic Information	Specify the Object Manager component for the selected application.
Language	Applications > Language > Basic Information	Displays the language for the selected application.
Request Start Command	Applications > Language > Basic Information	Specify the request start command for the selected application.
Configure EAI HTTP Inbound Transport	Applications > Language > Basic Information	Specify whether to configure the EAI HTTP Inbound Transport. Choose this option for EAI applications only, and not for any other applications.
Configure Anonymous Pool	Applications > Language > Basic Information This option appears if you selected Configure EAI HTTP Inbound Transport.	Specify whether to configure the anonymous pool for EAI requests. For more information, see *Siebel Performance Tuning Guide* .
Anonymous Pool Size	Applications > Language > Basic Information This option appears if you selected Configure EAI HTTP Inbound Transport and Configure Anonymous Pool.	Specify the size of the anonymous pool for EAI requests. For more information, see *Siebel Performance Tuning Guide* .

Setting in Siebel Management Console	Section (Under Create Profile)	Comment or Description
Maximum Retry for Processing EAI-SOAP Request	Applications > Language > Basic Information This option appears if you selected Configure EAI HTTP Inbound Transport.	Specify the maximum number of retries for processing SOAP requests for EAI.
No Session Preference in EAI-SOAP	Applications > Language > Basic Information This option appears if you selected Configure EAI HTTP Inbound Transport.	Specify whether the session preference is required in EAI-SOAP.
Configure Siebel Mobile	Applications > Language > Mobile	Specify whether to configure Siebel Mobile applications.
Application Display Name	Applications > Language > Mobile This option appears if you selected Configure Siebel Mobile.	Specify the display name of the Siebel Mobile application.
Application Display Order	Applications > Language > Mobile This option appears if you selected Configure Siebel Mobile.	Specify the display order of the Siebel Mobile application.
Application Icon Name	Applications > Language > Mobile This option appears if you selected Configure Siebel Mobile.	Specify the icon name of the Siebel Mobile application.
Active Session Timeout Value	Applications > Language > Enhanced Authentication **Note:** Use the Enhanced Authentication settings to optionally configure authentication for individual applications, if you require override settings different from the defaults specified under Basic Information and Authentication.	Specify the total number of seconds that a session can remain inactive before the user is logged out and the session is closed. Default value: 900 (seconds)
Login Session (guest session) Timeout Value	Applications > Language > Enhanced Authentication	Specify the timeout value for the login session. Default value: 300 (seconds)

Setting in Siebel Management Console	Section (Under Create Profile)	Comment or Description
Method to Check Server Availability	Applications > Language > Enhanced Authentication	Specify the method to use for checking server availability. Default value: UpdatePrefMsg
Command to Check Server Availability	Applications > Language > Enhanced Authentication	Specify the command to use for checking server availability. Default value: HeartBeat
Session Token Usage Duration	Applications > Language > Enhanced Authentication	Specify the usage duration for the session token. Default value: 2880 (minutes)
Session Token Timeout Value	Applications > Language > Enhanced Authentication	Specify the timeout value for the session token. Default value: 900 (seconds)
Configure Web Single Sign-On (Web SSO)	Applications > Language > Enhanced Authentication	Specify whether to use Web single sign-on.
Anonymous User Name	Applications > Language > Enhanced Authentication	Specify the Siebel user name that starts the anonymous session for the specified application. For example: GUESTCST For more information about anonymous users, see *Requirements for Configuring Anonymous Users for the Siebel Application Interface*.
Anonymous User Password	Applications > Language > Enhanced Authentication	Specify the password for the anonymous user.
Trust Token	Applications > Language > Enhanced Authentication This option appears if you selected Configure Web Single Sign-On.	Specify the trust token to use for Web SSO authentication.
User Specification	Applications > Language > Enhanced Authentication This option appears if you selected Configure Web Single Sign-On.	Specify the user specification to use for Web SSO authentication.
Resource Type	REST Inbound Resources > Data	In this section, you optionally provide alias names to the REST parameters. Specify the resource type for the REST resource parameter: Data, Service, or Workspace.
Method Name	REST Inbound Resources > Data > REST Resource Parameter List > Query	Specify the method name: Query (for example).

Setting in Siebel Management Console	Section (Under Create Profile)	Comment or Description
Name Alias	REST Inbound Resources > Data > REST Resource Parameter List > Query > Parameter List	For the current method name (such as Query), specify each REST resource parameter name for which you want to specify an alias, and specify the alias. For example, for the REST resource parameter named PageSize, you can configure an alias, such as Limit, as shown: Name=PageSize Alias=Limit Or, for the StartRowNumber parameter, you can specify an alias like Offset, as shown: Name=StartRowNumber Alias=Offset
Language	Other Information > SWE	Provide the default language for Siebel CRM.
HTTP-POST Request Size	Other Information > SWE	Specify the maximum size, in bytes, of HTTP POST requests from the Siebel Application Interface.
Seed File Location	Other Information > SWE	Specify the location of the seed file. (This setting is available for future use.)
Monitor Sessions	Other Information > SWE	Specify whether to gather statistics on all of the current sessions and report them in the Siebel Application Interface statistics (stats) page. For more information about this page, see *Siebel System Monitoring and Diagnostics Guide* .
Collect Application-Specific Statistics	Other Information > SWE	Specify whether to enable the collection of application-specific statistics.

Deploying the Siebel Application Interface

This procedure describes the steps for deploying a specific installed instance of Siebel Application Interface. You perform the same procedure for each installed instance. This task is part of *Configuring the Siebel Application Interface*.

This task is required both for new installations and for migration installations from a previous release, because the migration installation task does not migrate the prior configuration. You might also choose to install and deploy a new Siebel Application Interface.

Before you can deploy an installed instance of Siebel Application Interface, you must have already configured and deployed at least one Siebel Server, as described in *Configuring the Siebel Server*.

Note: After deployment, the Siebel Application Interface profile is in a read-write state. You can update configuration settings and save the profile to propagate the updates to the Siebel Application Interface.

Related Topics

Additional Tasks for Migration Installations

About Configuring Siebel CRM

Running the Siebel Management Console

Creating a Siebel Application Interface Profile

To deploy the Siebel Application Interface

1. Run the Siebel Management Console, as described in *Starting the Siebel Management Console*.
2. Click Siebel Deployment in the navigation menu.
3. In the overview area, click the plus sign (+) and then select Application Interface to add a new instance to your Siebel deployment.

   The Siebel Application Interface must already have been installed.
4. For Host Name:HTTPS Port, specify the host name and HTTPS redirect port for this instance of Siebel Application Interface. You specified this HTTPS redirect port during Siebel Application Interface installation.
5. For Profile, select the profile to use for this deployment of Siebel Application Interface.

   You created this profile in *Creating a Siebel Application Interface Profile*.
6. For Action, specify whether you will deploy this Siebel Application Interface or stage it for later deployment:
   - If you want to stage this Siebel Application Interface for later deployment, then click Staging.
   - If you want to deploy this Siebel Application Interface to your Siebel deployment, then click Deploy.
7. For Application Interface Node Name, specify the node name of this Siebel Application Interface.
8. For Application Interface Description, specify an optional description of this Siebel Application Interface.
9. To continue, click Submit. Otherwise, click Cancel.

   After the configuration has been submitted, the state of this Siebel Application Interface deployment item changes to one of the following states, based on your selection in Step 6 : Staging or Deployment in Progress.
   - If the deployment completes successfully, then, when you refresh the configuration, the state changes to Deployed. The Siebel Application Interface starts automatically.
   - If the deployment fails, then the state changes to Deployment Failed.
   - For a deployment in the Staging state, you can deploy it by either clicking the Deploy icon or by setting the Action to Deploy and resubmitting the deployment.
10. (Optional) View log data about this Siebel Application Interface deployment.

Configuring the Siebel Properties File

The following parameters may be found in the `siebel.properties` file:

- `siebel.conmgr.poolsize`= min and max recommended value : Defines the number of socket connections being readily maintained per MTServer under an OM. The default value of 2 suffices for most usage requirements and tuning might be required only when issues are faced for throughput or JDB related connection failures are seen in siebel.log under `\applicationcontainer_external\logs`.
- `siebel.conmgr.sesstimeout`= recommended value: This parameter is equivalent to the OM Task session timeout value on the Java side. This parameter need not be set for UI, REST and SOAP channels at

`\applicationcontainer_external`, as the value is inherited from SMC and is assigned at runtime. Any setting of the parameter in the `siebel.properties` file will override the SMC configured session timeout and applied for all object managers being connected.

The Siebel.properties above are used from the following location:

- `\applicationcontainer_external\lib` (The siebel.properties persist in an update.)
- `\applicationcontainer_external\webapps\siebel\WEB-INF\classes` (The siebel.prorperties will be overwritten in an update and hence need to be backed-up before patch upgrade).

Configuring the Siebel Gateway Cluster

After you have configured the primary Siebel Gateway and installed Siebel Gateway on additional nodes, you can optionally run the Siebel Management Console and perform tasks to configure the Siebel Gateway cluster, as described in this topic. This topic includes the following information:

- *About Siebel Gateway Clustering*
- *Creating a Siebel Gateway Cluster*
- *Configuration Settings for a Siebel Gateway Cluster Profile*
- *Deploying the Siebel Gateway Cluster*
- *Changing the Primary Nodes for a Siebel Gateway Cluster*
- *Restoring the Siebel Gateway Cluster Topology*

Note: The type of clustering described in this chapter applies only to the Siebel Gateway and is not available for other modules such as Siebel Server.

For more information about the activities that you can perform in the Siebel Management Console, including viewing configuration log data, see *Supported Activities in Siebel Management Console*.

This task is an optional step in:

- *Roadmap: Installing Siebel CRM for a New Deployment*
- *Roadmap: Installing Siebel CRM in a Migration Installation Case (Existing Installation of Siebel CRM 16.x or Earlier)*
- *Roadmap: Installing Siebel CRM in an Upgrade Case (Existing Database Requiring Full Upgrade)*

About Siebel Gateway Clustering

Siebel CRM supports an optional native clustering feature for Siebel Gateway to provide high availability benefits to Siebel CRM customers. This feature works at the software level and is the preferred and recommended approach for clustering the Siebel Gateway. This topic is part of *Configuring the Siebel Gateway Cluster*.

The clustering feature supports both the Siebel Gateway service (application container) and the Siebel Gateway registry (Apache ZooKeeper). You might choose to use Siebel Gateway clustering only for your production environment, for example. Further, you can use clustering for only the Siebel Gateway service, or only the Siebel Gateway registry. It is recommended to configure clustering for both the Gateway service and the Gateway registry. Clustering for the Siebel Gateway registry uses functionality native to Apache ZooKeeper, which is described in documentation for this third-party product.

For this feature, Siebel Management Console supports a new profile type, the Siebel Gateway cluster profile, and a new step to deploy the cluster profile.

Overview of Configuring Siebel Gateway Clustering

To use Siebel Gateway clustering, you perform the following tasks:

1. Install at least three instances of Siebel Gateway for Siebel CRM on different nodes. You can collocate a Siebel Gateway with a Siebel Server, for example, as noted in *Installing Siebel CRM Using the User Interface*.

   In a migration installation case, you perform a migration installation for the existing installed Siebel Gateway. Then you perform new installations for all other Siebel Gateway nodes that you require.
2. (New deployment only) Configure and deploy the primary Siebel Gateway node in Siebel Management Console, as described in *Configuring the Siebel Gateway and Security*.

   The first Siebel Gateway installed and configured is the primary node, unless you later changed the primary node to a different instance. This task does not apply in a migration installation case, because the existing Siebel Gateway is migrated and serves as the primary Siebel Gateway.
3. Start the Siebel Management Console.
4. Configure a Siebel Gateway cluster profile in Siebel Management Console, as described in *Creating a Siebel Gateway Cluster*.

   In the cluster profile, you specify values for Registry Client Port, Registry Follower Port, and Registry Leader Port. For Registry Client Port, you specify the same Siebel Gateway registry port number that you specified when you configured the primary Siebel Gateway, as described in *Configuring the Siebel Gateway and Security*. The Registry Follower Port and Registry Leader Port are used for internal communication between Siebel Gateway nodes.
5. (Required as precaution) Back up the Siebel Gateway registry, as described in *Siebel System Administration Guide* .
6. Deploy the Siebel Gateway cluster profile, specifying all participating nodes, as described in *Deploying the Siebel Gateway Cluster*.

Siebel Gateway Clustering Requirements and Operations

The specific requirements for Siebel Gateway clustering differ for the Siebel Gateway service and the Siebel Gateway registry. Optionally, you can deploy the Siebel Gateway service cluster and registry cluster on different nodes.

- Siebel Gateway service clustering requires a minimum of two nodes. You can include all the same nodes that you include for the registry cluster, or include only a subset of these nodes.
- Siebel Gateway registry clustering requires a minimum of three nodes and an odd number of nodes: three, five, and so on.

Note: At any time after the primary Siebel Gateway has been configured in Siebel Management Console and the participating nodes have been installed, you can configure and deploy the Siebel Gateway cluster or change an existing cluster, such as to add or remove nodes or change the primary service node and primary registry node. In general, it is recommended to deploy the Siebel Gateway cluster before you configure the Siebel Enterprise, Siebel Server, and so on. If you deploy the Siebel Gateway cluster after you configure the rest of the Siebel environment, then afterward you must restart the application containers for all of the other services.

After cluster deployment, all applicable nodes function as part of the cluster, for the Siebel Gateway service and the Siebel Gateway registry, as applicable. The Siebel Deployment screen in Siebel Management Console shows all of the components of the Siebel Gateway cluster, within the overall deployment hierarchy. The item Gateway Cluster has child items Gateway Service Cluster and Gateway Registry Cluster, each of which has child items representing the

participating nodes. Deployment status is shown for each cluster node. The primary node is highlighted, for your reference. If deployment does not succeed for the minimum number of nodes required for each cluster, then cluster deployment fails and the system uses the primary Siebel Gateway.

Siebel Application Interface and Siebel Server function as clients for the Siebel Gateway service. When a Siebel Gateway service cluster is deployed, then Siebel Application Interface or Siebel Server can communicate with any of the nodes in the Siebel Gateway service cluster, and route requests to each available node, in a round-robin manner.

Similarly, the Siebel Gateway service that receives a request functions as a client for the Siebel Gateway registry. When a Siebel Gateway registry cluster is deployed, then the Siebel Gateway service can communicate with any of the nodes in the Siebel Gateway registry cluster, and routes each request to an available node, in a manner determined by ZooKeeper.

If an individual node goes down within the Siebel Gateway cluster, then the Siebel Application Interface or Siebel Server client connection switches to another available node, within the tolerance allowed by the cluster. After any Siebel Gateway cluster node goes down, when it is restarted, the node will again participate in the cluster.

- For a Siebel Gateway service cluster, all but one node can go down, and the Siebel Gateway service will still function.
- However, for the Siebel Gateway registry cluster, among those nodes configured as part of the cluster, a simple majority of running instances is required in order for the registry cluster to function: for example, two out of three nodes, or three out of five nodes.

If the Zookeeper Siebel Gateway cluster deployment fails (for example, if two out of three Siebel Gateway registry cluster nodes go down), then the system reverts to the non-clustered architecture: Siebel Application Interface and Siebel Server connect to the primary Siebel Gateway node, assuming it is running. (In this case, you can change the primary service node or primary registry node to a node that is functional if the system is no running.)

The Siebel Gateway cluster uses an active-active model, in which actions can originate in any available node. For the Siebel Gateway registry cluster, updates are replicated to the other nodes, thus keeping all of the cluster nodes synchronized. The active-active model enables the cluster nodes to participate in load balancing of Siebel Gateway activities.

If multiple network interface controllers are used in your deployment environment, then advanced DNS and IP networking concepts must be applied in your cluster configuration. To avoid DNS routing conflicts that could occur, you might need to specify IP addresses rather than host names.

To configure the gateway cluster with multiple networks

1. Follow the instructions described in *Configuring the Siebel Gateway Cluster*.

   Note: This configuration only applies to Siebel 21.4 and above. It is mentioned on Bookshelf to take a backup copy of the Siebel Gateway registry before starting the Gateway Cluster configuration. A successful Gateway Cluster deployment depends on all used TCP ports be open bi-directionally on the firewall.

2. After deploying the Gateway cluster successfully on the Siebel Management Console (SMC), edit the registry.cfg located under ses/gtwysrvr/registry/conf directory to extend the leader or follower port definition by adding a second NIC. The following example shows how a non-modified registry.cfg looks like:

```
#Fri Nov 19 20:54:07 UTC 2021
autopurge.purgeInterval=1
initLimit=10
syncLimit=5
autopurge.snapRetainCount=10
maxClientCnxns=10000
clientPort=2330
```

```
tickTime=2000
multiAddress.reachabilityCheckEnabled=false
dataDir=/refresh/siebel/ses/gtwysrvr/registry/conf
admin.enableServer=false
server.1=cgw1.domain-A:2328:2329
server.2=cgw2.domain-A:2328:2329
server.3=cgw3.domain-A:2328:2329
```

The following example shows how the modified registry.cfg would look like:

```
#Fri Nov 19 20:54:07 UTC 2021
autopurge.purgeInterval=1
initLimit=10
syncLimit=5
autopurge.snapRetainCount=10
maxClientCnxns=10000
clientPort=2330
tickTime=2000
multiAddress.reachabilityCheckEnabled=false
dataDir=/refresh/siebel/ses/gtwysrvr/registry/conf
admin.enableServer=false
server.1=cgw1.domain-A:2328:2329|cgw1.domain-B:2328:2329
server.2=cgw2.domain-A:2328:2329|cgw2.domain-B:2328:2329
server.3=cgw3.domain-A:2328:2329|cgw3.domain-B:2328:2329
```

3. _JAVA_OPTIONS environment variable is required to enable the Zookeeper feature. On Unix or Linux operating systems, append the following export command to the siebenv.sh shell script located under the gtwysrvr directory.

   ```
   export _JAVA_OPTIONS="-Dzookeeper.multiAddress.enabled=true"
   ```

   On the Windows operating system, define the environment variable as a system environment variable.
4. Implement the above changes manually on all Gateway Cluster nodes.
5. Restart the Gateway Registry service on all nodes for the changes to take place.

Note: A proper Gateway Cluster setup depends on the leader and follower TCP ports to be open bi-directionally in the firewall rules.

Related Topics

About Configuring Siebel CRM

Running the Siebel Management Console

Configuring the Siebel Gateway and Security

Changing the Primary Nodes for a Siebel Gateway Cluster

Removing the Siebel Gateway

Related Books

Siebel Deployment Planning Guide

Siebel System Administration Guide

Creating a Siebel Gateway Cluster

This procedure describes how to configure a Siebel Gateway cluster profile. You can create multiple profiles, but you can deploy only one Siebel Gateway cluster in your Siebel deployment. This task is part of *Configuring the Siebel Gateway Cluster*.

Related Topics

Running the Siebel Management Console

About Siebel Gateway Clustering

Deploying the Siebel Gateway Cluster

To create a Siebel Gateway Cluster profile

1. Run the Siebel Management Console, as described in *Starting the Siebel Management Console*.
2. Click Profiles in the navigation menu in the side panel, then click Gateway Cluster.

   Existing Siebel Gateway cluster profiles are listed, if any.
3. Click the plus (+) icon to add a new Siebel Gateway cluster profile, or click the Clone button to clone an existing profile.
4. Specify a name for the profile.
5. Specify additional settings, as shown in *Configuration Settings for a Siebel Gateway Cluster Profile*.
6. When you have specified all applicable settings, click Submit to save your changes to the profile.

Configuration Settings for a Siebel Gateway Cluster Profile

The following table describes the settings that are requested in the Siebel Management Console for configuring a new Siebel Gateway cluster profile or modifying an existing Siebel Gateway cluster profile. Required fields display an asterisk (*) next to the field label. Pointing to the option displays tooltip text that provides a brief description of the field. This topic is part of *Configuring the Siebel Gateway Cluster*.

Setting in Siebel Management Console	Section (Under Create Profile)	Comment or Description
Registry Client Port	Gateway Cluster Settings	Specify the port number to use for the Siebel Gateway registry on all nodes. You can specify a new port number or specify the same Siebel Gateway registry port number that you specified when you configured the primary Siebel Gateway, as described in *Configuring the Siebel Gateway and Security*.
Registry Follower Port	Gateway Cluster Settings	The Registry Follower Port and Registry Leader Port are used for internal communication between nodes participating in Siebel Gateway registry clustering.

Setting in Siebel Management Console	Section (Under Create Profile)	Comment or Description
Registry Leader Port	Gateway Cluster Settings	The Registry Follower Port and Registry Leader Port are used for internal communication between nodes participating in Siebel Gateway registry clustering.

Deploying the Siebel Gateway Cluster

This procedure describes how to deploy the Siebel Gateway cluster. You can deploy only one Siebel Gateway cluster in your Siebel deployment. This task is part of *Configuring the Siebel Gateway Cluster*.

> **Note:** You must back up the registry of the primary Siebel Gateway before you deploy the Siebel Gateway cluster. If a failure is observed, then you must restore the Siebel Gateway registry, after which you can make another attempt to deploy the Siebel Gateway cluster. Restoring the Siebel Gateway registry requires you to restart your CRM deployment. For more information about backing up and restoring the Siebel Gateway registry and about stopping and starting the Siebel CRM deployment, see *Siebel System Administration Guide* . See also *Stopping and Starting the Siebel Application Container*.

Related Topics

About Configuring Siebel CRM

Running the Siebel Management Console

About Siebel Gateway Clustering

Creating a Siebel Gateway Cluster

To deploy the Siebel Gateway cluster

1. (Required as precaution) Back up the Siebel Gateway registry, as described in *Siebel System Administration Guide* .
2. Run the Siebel Management Console, as described in *Starting the Siebel Management Console*.
3. Click Siebel Deployment in the navigation menu.
4. In the overview area, click the plus sign (+) and then select Gateway Cluster to add a new instance to your Siebel deployment.

   Siebel Gateway must already have been installed, on a sufficient number of nodes. You can deploy a Siebel Gateway cluster only once, but you can then modify the cluster if necessary, such as to add nodes.
5. For Profile, select the profile to use for this deployment of Siebel Gateway cluster.

   You created this profile in *Creating a Siebel Gateway Cluster*.
6. Under Gateway Service Cluster, for each installed instance that will participate in this cluster, specify the host name and port number.
7. Under Gateway Registry Cluster, for each installed instance that will participate in this cluster, specify the host name and port number.

8. For Action, specify whether you will deploy this Siebel Gateway cluster or stage it for later deployment:
   - If you want to stage this Siebel Gateway cluster for later deployment, then click Staging.
   - If you want to deploy this Siebel Gateway cluster to your Siebel deployment, then click Deploy.
9. To continue, click Submit. Otherwise, click Cancel.
   After the configuration has been submitted, the state of this Siebel Gateway cluster deployment item changes to one of the following states, based on your selection in Step 8: Staging or Deployment in Progress.
   - If the deployment completes successfully, then the state changes to Deployed. The Siebel Gateway cluster starts automatically.
   - If the deployment fails, then the state changes to Deployment Failed.
   - For a deployment in the Staging state, you can deploy it by either clicking the Deploy icon or by setting the Action to Deploy and resubmitting the deployment.
10. (Optional) View log data about this Siebel Gateway cluster deployment.

Changing the Primary Nodes for a Siebel Gateway Cluster

After you have configured and deployed the primary Siebel Gateway and additional nodes for a Siebel Gateway cluster, you can optionally change which service and registry nodes serve as the primary nodes for the cluster. This task is part of *Configuring the Siebel Gateway Cluster*.

Note the following requirements for using this feature:

- You can change the primary node for the Siebel Gateway service, for the Siebel Gateway registry, or both.
- You can set the primary service node and primary registry node on two different host computers.
- For either operation, the current primary node does not have to be running in order to specify a different node as the primary. The node you are specifying as the new primary must be running.
- If you attempt to delete the cluster, this action removes all of the nodes from the cluster hierarchy except the primary service node and primary registry node. You must add additional nodes to the Siebel Gateway cluster before it can function as a cluster again.
- Before you can restore a prior release of Siebel CRM, the primary service node and registry node for the Siebel Gateway cluster must be located on the same host computer. Also validate that the `applicationcontainer_internal\webapps\gateway.properties` file for this primary service node (and for no other node) includes the setting Primed=True. This primary service node must be operational in order to serve as the primary Siebel Gateway node after restoring a prior release. Otherwise, cluster modification might fail and the Siebel Gateway service might go down. See also *Requirements for Uninstalling Siebel CRM*.

To change the primary node for the Siebel Gateway service

1. In the Siebel Deployment screen in Siebel Management Console, navigate to the Hierarchy - Gateway Service Cluster view.
   For more information about this screen, see *Siebel Management Console Screens*. Under Gateway Service, the primary service node within this cluster is shown with bold text.
2. Select the node that you want to designate as the primary service node.
3. Click the Primary button.
   A dialog box prompts you to confirm that you want to designate this node as the primary service node for the cluster.

4. Click OK to confirm this change.

   An alert box indicates that the change has been made.
5. Click OK again.

   The Siebel Deployment screen appears again, which includes the cluster for which you changed the service node.

To change the primary node for the Siebel Gateway registry

1. In the Siebel Deployment screen in Siebel Management Console, navigate to the Hierarchy - Gateway Registry Cluster view.

   For more information about this screen, see *Siebel Management Console Screens*. Under Gateway Registry, the primary registry node within this cluster is shown with bold text.
2. Select the node that you want to designate as the primary registry node.
3. Click the Primary button.

   A dialog box prompts you to confirm that you want to designate this node as the primary registry node for the cluster.
4. Click OK to confirm this change.

   An alert box indicates that the change has been made.
5. Click OK again.

   The Siebel Deployment screen appears again, which includes the cluster for which you changed the registry node.

Related Topics

Siebel Management Console Screens

Restoring the Siebel Gateway Cluster Topology

Requirements for Uninstalling Siebel CRM

Restoring the Siebel Gateway Cluster Topology

Your Siebel Gateway cluster topology could, in rare situations, get out of sync with the Siebel Application Interface, leading to an inability to connect to a Siebel Gateway. This problem can occur only when you have taken down the primary Siebel Gateway service node and removed it from a cluster. In order to prevent a situation where no valid Siebel Gateway connection information is available following a Siebel CRM update installation, an upgrade sequence is provided that resolves the described problem. This task is part of *Configuring the Siebel Gateway Cluster*.

The file cgclientstore.dat, located in `SIEBEL_AI_ROOT\applicationcontainer_external\webapps` on the Siebel Application Interface, is a runtime file that is created when the Siebel Gateway cluster is first created. In normal operation, this file contains the names of all of the Siebel Gateway service nodes (Tomcat instances) that are currently running, and provides connection information for a running instance in response to requests. These three nodes would include, for example:

- The primary Siebel Gateway service node, from which the Siebel Gateway was bootstrapped. For example, this node might be GTWY1.

- Additional Siebel Gateway service nodes that were added as part of a cluster. For example, these nodes might be GTWY2 and GTWY3.

The applicationinterface.properties file, located in `SIEBEL_AI_ROOT\applicationcontainer_external\webapps` on the Siebel Application Interface, includes the following entry, where GTWY1 represents the primary Siebel Gateway service node. The CGHostURI parameter setting provides a backup source for connection information for a running Siebel Gateway service node, in case cgclientstore.dat is empty. The following is an example value for CGHostURI, allowing connection to GTWY1:

```
CGHostURI=https://GTWY1.us.oracle.com:4431/siebel/v1.0/cloudgateway
```

Assume, for this troubleshooting scenario, that the customer has taken down GTWY1 and removed it from the cluster. In that case, CGHostURI indicates an invalid Siebel Gateway service node. When you update to the current Siebel CRM release, you must follow the upgrade sequence here to avoid a case where cgclientstore.dat is empty and CGHostURI points to an invalid node.

1. Install Siebel CRM for all Siebel Gateway nodes before you update any other modules.
2. Stop the Siebel Gateway service instances that start automatically following installation.
3. Start the Siebel Gateway registry instances.
4. Start the Siebel Gateway service instances.
5. Install Siebel CRM for all Siebel Application Interface instances, Siebel Servers, and so on.

When you perform the update in this sequence, then cgclientstore.dat is repopulated with connection information for running instances of Siebel Gateway service node, in response to requests. For more information about starting and stopping the Siebel Gateway, see *Starting the Siebel Gateway*. See also the information about starting and shutting down a Siebel CRM deployment, located in *Siebel System Administration Guide* .

Related Topics

Changing the Primary Nodes for a Siebel Gateway Cluster

Related Books

Siebel System Administration Guide

Configuring the Siebel Migration Application

The Siebel Migration application, a Web-based tool for migrating Siebel Repositories and seed data and performing related tasks, is provided with the Siebel Application Interface installation. A variety of configuration options are provided to support different ways of using the Siebel Migration application.

To configure the Siebel Migration application, perform the following tasks:

1. *Creating a Siebel Migration Profile*
2. *Deploying the Siebel Migration Application*

For more information about the activities that you can perform in the Siebel Management Console, including viewing configuration log data, see *Supported Activities in Siebel Management Console*. For more information about the role of the Siebel Migration application in the overall deployment process, see *Siebel Database Upgrade Guide* .

This task is a step in:

- *Roadmap: Installing Siebel CRM for a New Deployment*
- *Roadmap: Installing Siebel CRM in a Migration Installation Case (Existing Installation of Siebel CRM 16.x or Earlier)*

- *Roadmap: Installing Siebel CRM in an Upgrade Case (Existing Database Requiring Full Upgrade)*

Related Topics

About Configuring Siebel CRM

Running the Siebel Management Console

Related Books

Siebel Database Upgrade Guide

Siebel Security Guide

Creating a Siebel Migration Profile

This procedure describes how to configure a Siebel Migration profile. You can create multiple profiles. This task is part of *Configuring the Siebel Migration Application*.

For more information about the activities that you can perform in the Siebel Management Console, including viewing configuration log data, see *Supported Activities in Siebel Management Console*.

> **Note:** Several existing settings for Siebel Migration profiles no longer apply, and a new setting is provided. You must take these changes into account when installing and using Siebel CRM 20.10 or later to update an existing release, or when restoring a prior release. After updating, if you update existing Siebel Migration profiles, then values for several database-related settings are lost and you can add a value for the new setting Migration REST Endpoint for Authentication and Data Store. Similarly, after restoring a prior release, if you had created or updated a Siebel Migration profile to include the new setting Migration REST Endpoint for Authentication and Data Store, then you must update the Siebel Migration profile after restoring the prior release to restore the values for the database-related settings that became obsolete as of Siebel CRM 20.10: Host Name, Port Number, Table Owner, User Name, Password, RDBMS Platform, Service Name, Database Instance, Database Name, and Authentication Host. Any profile that you did not update after installing the update retains the pre-20.10 settings and can be used after restoring the prior release. See also *Configuration Settings for a Siebel Migration Profile*.

Related Topics

Running the Siebel Management Console

Deploying the Siebel Migration Application

To create a Siebel Migration profile

1. Run the Siebel Management Console, as described in *Starting the Siebel Management Console*.
2. Click Profiles in the navigation menu in the side panel, then click Migration.

   Existing Siebel Migration profiles are listed, if any.
3. Click the plus (+) icon to add a new Siebel Migration profile, or click the Clone button to clone an existing profile.
4. Specify a name for the profile.
5. Specify additional settings, as shown in *Configuration Settings for a Siebel Migration Profile*.
6. When you have specified all applicable settings, click Submit to save your changes to the profile.

Adding a Migration Profile Using REST

You can also create a Migration Profile using an inbound REST request. To do this send a request in the following format, filling in your custom values for the parameters. This is an example of the body for the REST request.

The following details are for a request to insert a Migration Profile:

URI: `https://{hostname}:{port}/siebel/v1.0/migration/`

- **HTTP Method:** `PUT`
- **Content-Type:** `application/json`
- **Authorization:** `Basic`

Body:

```
{
 "MigrationConfigParams":
 {
 "SiebelRESTEndpoint":"https://siebel.demo.tst/siebel/v1.0",
 "AuthenticationType":"Basic",
 "LogLevel":"Error",
 "Timeout":300,
 "SleepTime":15,
 "SiebelApplicationName":"eai",
 "Language":"enu",
 "PackageLocation":""
 },
"Profile":{"ProfileName":"test_profile"}
}
```

Deploying a Migration Profile Using REST

In addition to creating a Migration Profile using a REST request, you can also deploy the profile you created using a REST request. Here is an example of the body of the request.

- **URI:** `https://{hostname}:{port}/siebel/v1.0/migration/`
- **HTTP Method:** `PUT`
- **Content-Type:** `application/json`
- **Authorization:** `Basic`

Body

```
{
 "DeploymentInfo":
 {
 "PhysicalHostIP":"siebel.demo.tst:443",
 "ProfileName":"test_profile",
 "Action":"Deploy"
 },
 "MigrationDeployParams":
 {
 "SiebelMigration":"migration",
 "MigrationDesc":"migration"}
}
```

Configuration Settings for a Siebel Migration Profile

The following table lists the settings requested in the Siebel Management Console for configuring the Siebel Migration profile or modifying an existing Siebel Migration profile. This topic is part of *Configuring the Siebel Migration Application*.

Required fields display an asterisk (*) next to the field label. For most of the options, pointing to the option displays tooltip text that provides a brief description of the field.

Setting in Siebel Management Console	Section (Under Create Profile)	Comment or Description
Migration REST Endpoint for Authentication and Data Store	Connectivity and Authentication	Specify the REST endpoint URL for the Siebel Migration application. This URL is used for authenticating the Siebel Migration user and for storing the Siebel Migration application data. The default URL format is the Siebel Management Console URL. Update the default value to match the value shown here. https://AI_HOST:AI_PORT/siebel/v1.0 In this URL, AI_HOST is the Application Interface host name and AI_PORT is the port number. The keyword *siebel* can optionally be replaced, as described in *Customizing URLs for Siebel CRM Applications*. This setting is new as of Siebel CRM 20.10. For more information, see *Creating a Siebel Migration Profile*.
Authentication Type	Authentication	Specify the authentication type for the Siebel Migration application. Specify one of the following: • Basic Authentication • Single Sign-On
User Specification	Authentication This option appears if you selected Single Sign-On Authentication.	Provide the user specification for SSO authentication.
Assertion Specification	Authentication This option appears if you selected Single Sign-On Authentication.	Provide the assertion specification for SSO authentication.
Identity Provider Logoff URL	Authentication This option appears if you selected Single Sign-On Authentication.	Provide the identity provider logoff URL for SSO authentication.

Setting in Siebel Management Console	Section (Under Create Profile)	Comment or Description
Parameter Name for Identity Provider Logoff Return URL	Authentication This option appears if you selected Single Sign-On Authentication.	Provide the parameter name for identity provider logoff return URL for SSO authentication.
Log Level	Logging Information	Specify the logging level for the Siebel Migration application. Select one of the following options: Fatal, Error, Debug, Trace
Timeout	Other Information	Specify the session timeout in seconds for the Siebel Migration application. Default: 30 (seconds)
Sleep Time	Other Information	Specify the sleep time in seconds for the Siebel Migration application. Default: 15 (seconds) **Note:** Do not change this value unless instructed to do so by Oracle for troubleshooting. Doing so may cause Migrations to fail as the poll time the application uses to check on progress and initiate next steps in the Migration may be too long.
Siebel Application Name for Data Administration	Other Information	Specify the Siebel application name that needs to be embedded in the Siebel Migration application.
Language	Other Information	Specify the language of the Siebel application that needs to be embedded in the Siebel Migration application.
Migration Package Location	Other Information	You can optionally specify a single shared migration package location that supports both synchronous and asynchronous migrations in your source and target environments. **Note:** More information about how this setting is used is provided in *About the Migration Package Location*.

About the Migration Package Location

You can optionally specify a single shared migration package location that supports both synchronous and asynchronous migrations in your source and target environments. This network file share path must be accessible to all of the connections that are registered in the Siebel Migration application. This topic is part of *Configuring the Siebel Migration Application*.

The Migration Package Location setting functions as follows:

- **Synchronous migration.** The Siebel Migration application exports source environment data into a file, copies the file into the specified migration package location, and then imports data from this file into the target environment.

  If you do not use this setting, then the Siebel Migration application uses the Migration File Transfer REST service to transfer the file automatically from the Siebel File System on the source environment into the corresponding location on the target.
- **Asynchronous migration.** The Siebel Migration application exports source environment data into a file and copies the file into the specified migration package location. The customer must start the import process to import data from this file into the target environment.

  If you do not use this setting, then you must copy the file manually from the Siebel File System on the source environment into the corresponding location on the target before you can import the data.

In some environments, restrictions might prevent you from using the Migration Package Location setting to specify a shared migration package location. In such cases, you might be required to use asynchronous migration and to copy the export file manually.

Because the Siebel Migration application uses the Siebel File System (`migration` subdirectory), it is strongly recommended to use different file system locations for the source and target environments, to avoid any file conflicts. For more information, see *Creating the Siebel File System*.

The Siebel service owner account and the Siebel Application Interface owner account must have read-write access to the Siebel File System and the Migration Package Location. See also *Creating the Siebel Service Owner Account*.

Deploying the Siebel Migration Application

This procedure describes the steps for deploying the Siebel Migration application. This task is part of *Configuring the Siebel Migration Application*.

This task is required for new installations where you choose to use the Siebel Migration application. It is also required for migration installations where you choose to use the Siebel Migration application, because this application did not exist prior to Siebel CRM 17.0 and there is no prior configuration to migrate.

In order to use the Siebel Migration application, the Remote component group must be enabled in the source environment and the EAI and Workflow component groups must be enabled in both the source and target environments. The workflow named File Transfer Workflow must be active in both the source and target environments. Synchronize the Siebel Server components on both environments and restart the entire Siebel CRM deployment. Before you execute the migration plan, make sure that the relevant server components are active that the Siebel Migration application uses to perform the migration. For more information about using the Siebel Migration application, see *Siebel Database Upgrade Guide* .

> **Note:** After deployment, the Siebel Migration profile is in a read-write state. You can update configuration settings and redeploy the profile to propagate the updates to the Siebel Migration application.

Related Topics

Running the Siebel Management Console

Creating a Siebel Migration Profile

To deploy the Siebel Migration application

1. Run the Siebel Management Console, as described in *Starting the Siebel Management Console*.
2. Click Siebel Deployment in the navigation menu.
3. In the overview area, click the plus sign (+) and then select Migration to add a new instance to your Siebel deployment.
4. For Host Name:HTTPS Port, specify the host name and HTTPS redirect port for this instance of Siebel Application Interface. You specified this HTTPS redirect port during Siebel Application Interface installation.
5. For Profile, select the profile to use for this deployment of the Siebel Migration application.
   You created this profile in *Creating a Siebel Migration Profile*.
6. For Action, specify whether you will deploy the Siebel Migration application or stage it for later deployment:
   - If you want to stage the Siebel Migration application for later deployment, then click Staging.
   - If you want to deploy the Siebel Migration application to your Siebel deployment, then click Deploy.
7. For Siebel Migration Name, specify a deployment name for this instance of the Siebel Migration application.
8. For Siebel Migration Description, specify an optional description of this instance of the Siebel Migration application.
9. To continue, click Submit. Otherwise, click Cancel.
   After the configuration has been submitted, the state of this Siebel Migration deployment item changes to one of the following states, based on your selection in Step 6: Staging or Deployment in Progress.
   - If the deployment completes successfully, then, when you refresh the configuration, the state changes to Deployed.
   - If the deployment fails, then the state changes to Deployment Failed.
   - For a deployment in the Staging state, you can deploy it by either clicking the Deploy icon or by setting the Action to Deploy and resubmitting the deployment.
10. (Optional) View log data about this Siebel Migration deployment.

Configuring the Siebel Deployment

System administrators can configure server elements in the Siebel CRM deployment by using the Configuration screen in the Siebel Management Console. Administrators can perform activities here that are equivalent to some of the activities that you traditionally perform in the Administration - Server Configuration screen in the Siebel application or using the Server Manager command-line utility. This functionality is provided as an alternative to the traditional methods.

The Configuration screen in the Siebel Management Console supports a subset of the functionality from the Administration - Server Configuration screen. For example, Siebel Management Console does not include the Job Templates feature or, under Enterprises, the Synchronize and System Alerts features.

Siebel System Administration Guide describes the existing server configuration functionality and includes information about relevant configuration settings and operations, some of which also apply to what you can do in the Siebel Management Console.

This topic includes the following information:

- *About the Configuration Screen*
- *Activities in the Configuration Screen*

- *Configuring an Element in the Configuration Screen*

Related Topics

About Configuring Siebel CRM

Running the Siebel Management Console

Managing the Siebel Deployment

Related Books

Siebel System Administration Guide

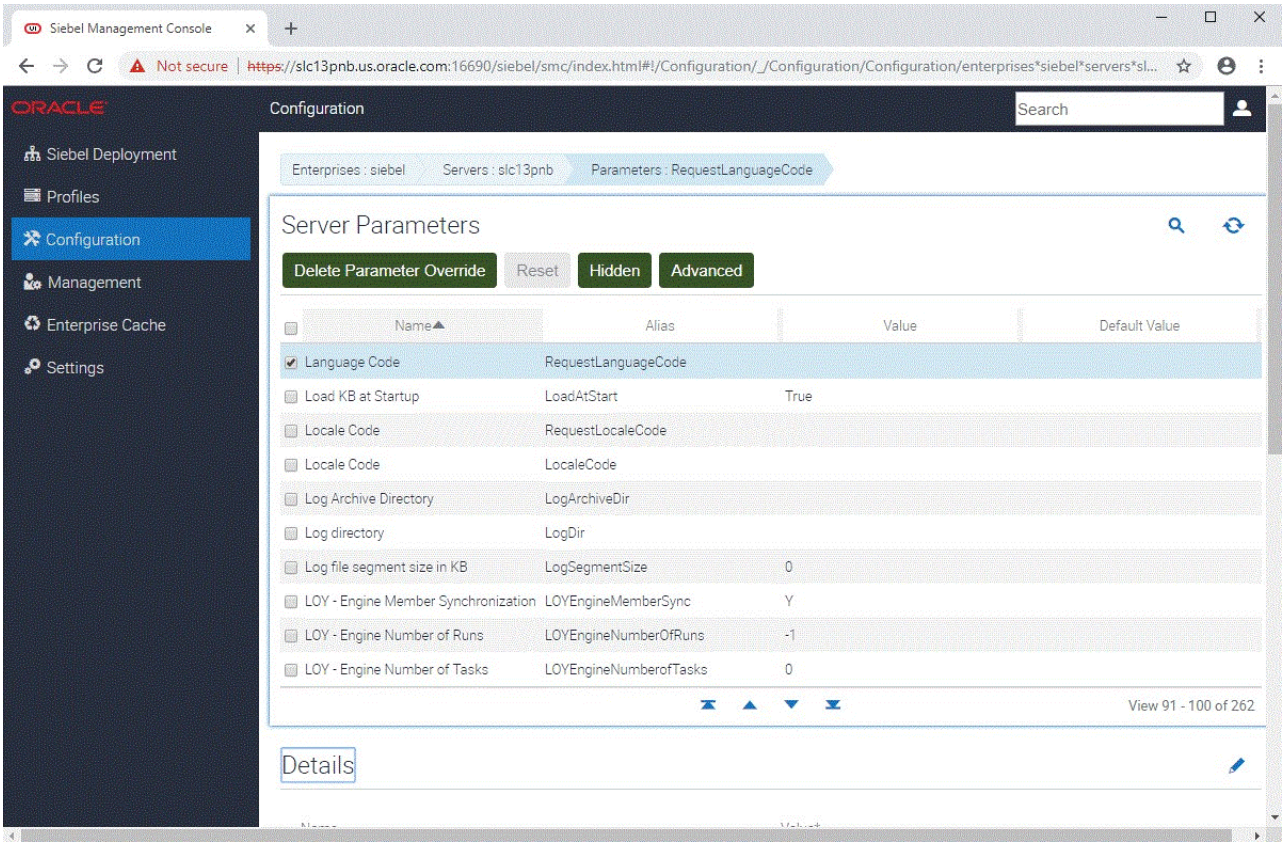
About the Configuration Screen

You can perform activities in the Configuration screen in Siebel Management Console at any time after you complete the required profile configuration and deployment tasks for Siebel deployment elements. The Configuration screen includes views that allow you to configure the Siebel enterprise and its component Siebel Servers, server components, parameters, and so on. This topic is part of *Configuring the Siebel Deployment*.

The default view in the Configuration screen identifies the current Siebel enterprise. The navigation bar in the header area of each view is a breadcrumb control that lets you navigate within the hierarchical elements of the Siebel enterprise. As shown in the following, you can select Component Definitions, Component Groups, Servers (for configuring a Siebel Server), Profile Configuration (for configuring an enterprise profile), or Parameters (for configuring enterprise parameters). When you have selected one of these options, you can navigate further into the hierarchy, such as to configure parameters for a particular Siebel Server or one of its server components. You can navigate up or down the hierarchy using this navigation bar.

Icons in the header area of each view or panel (list or form type) indicate operations that can be performed for the selected element or subelement (for example, Siebel Server parameters), such as to create, delete, or edit an instance of the element, or to query for the element you need. The available options vary by the particular context of the operations you are performing in this screen.

The following figure shows the Configuration screen in Siebel Management Console. You can use this screen to perform configuration tasks for the Siebel Enterprise, Siebel Server, server components, and related areas. The navigation menu, shown in the side panel, provides access to the Configuration screen and to other functionality of Siebel Management Console.



Activities in the Configuration Screen

The following table lists the primary configurable elements in the Configuration screen in the Siebel Management Console and summarizes some of the available configuration activities. This topic is part of *Configuring the Siebel Deployment*.

Configurable Element Under Siebel Enterprise	Available Activities
Component Definitions	Activities for component definitions include creating, deleting, updating, activating, deactivating, or synchronizing a component definition. Child elements: Parameters
Component Groups	Activities for component groups include creating, deleting, updating, enabling, or disabling a component group. Child elements: Component Definitions, Parameters
Servers	Activities for Siebel Servers include configuring parameters and other elements for a Siebel Server. Child elements: Components, Events, Parameters

Configurable Element Under Siebel Enterprise	Available Activities
Profile Configuration	Activities for enterprise profiles (named subsystems) include creating, deleting, updating, or configuring parameters for a profile. Child elements: Parameters
Parameters	Activities for enterprise parameters include configuring parameters or deleting parameter overrides.

Configuring an Element in the Configuration Screen

To configure Siebel enterprise elements in the Configuration screen in the Siebel Management Console, perform the following general steps. This task is part of *Configuring the Siebel Deployment*.

To configure an element in the Configuration screen

1. Run the Siebel Management Console, as described in *Starting the Siebel Management Console*.
2. Click Configuration in the navigation menu in the side panel.
3. For the current Siebel enterprise, navigate to one of the options available from the navigation bar.
   The options under the Siebel enterprise are shown in the table in *Activities in the Configuration Screen*.
4. Select, create, delete, or modify an item (such as an enterprise parameter, or a Siebel Server), as appropriate.
5. Optionally, from the navigation bar, navigate to one of the available child elements for the selected item (such as parameters for a Siebel Server), and perform configuration operations.
   The available child elements depend on the item selected in Step 4, and are also shown in the table in *Activities in the Configuration Screen*.
6. Select, create, delete, or modify an item, as appropriate.

Managing the Siebel Deployment

System administrators can manage server modules in the Siebel CRM deployment by using the Management screen in the Siebel Management Console. Administrators can perform activities here that are equivalent to some of the activities that you traditionally perform in the Administration - Server Management screen in the Siebel application or using the Server Manager command-line utility. This functionality is provided as an alternative to the traditional methods.

The Management screen in the Siebel Management Console supports a subset of the functionality from the Administration - Server Management screen.

Siebel System Administration Guide describes the existing server management functionality and includes information about relevant management settings and operations, some of which also apply to what you can do in the Siebel Management Console.

This topic includes the following information:

- *About the Management Screen*
- *Activities in the Management Screen*

- *Managing an Element in the Management Screen*

Related Topics

About Configuring Siebel CRM

Running the Siebel Management Console

Configuring the Siebel Deployment

Related Books

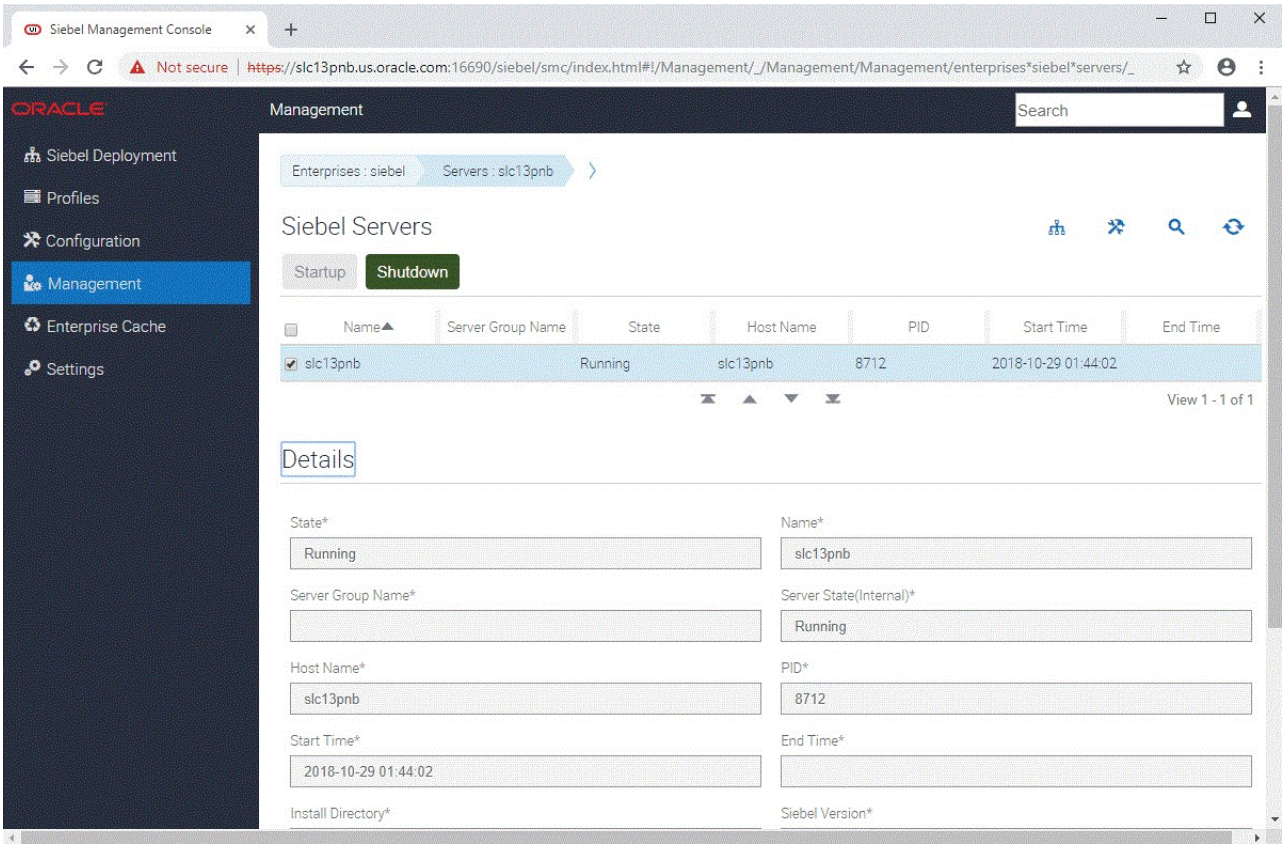
Siebel System Administration Guide

About the Management Screen

You can perform activities in the Management screen in Siebel Management Console at any time after you complete the required profile configuration and deployment tasks for Siebel deployment elements. The Management screen includes views that allow you to manage the Siebel Servers, server components, sessions, and tasks in your Siebel deployment. This topic is part of *Managing the Siebel Deployment*.

The default view in the Management screen identifies the current Siebel enterprise. The navigation bar in the header area of each view is a breadcrumb control that lets you navigate within the hierarchical elements of the Siebel enterprise. You can select Servers, Components, Sessions, or Tasks. When you have selected one of these options, you can navigate further into the hierarchy, such as to manage components for a particular Siebel Server, manage tasks for a particular component, and so on. You can navigate up or down the hierarchy using this navigation bar.

The following figure shows the Management screen in Siebel Management Console. The navigation menu, shown in the side panel, provides access to the Management screen and to other functionality of Siebel Management Console.



Activities in the Management Screen

The following table lists some of the management activities that you can perform using the Management screen in the Siebel Management Console. This topic is part of *Managing the Siebel Deployment*.

Manageable Element Under Siebel Enterprise	Management Activity
Servers	You can do the following: • Review the status of any items in the hierarchy (component groups, components, logs, sessions, statistics, and tasks). • Stop and start a Siebel Server. • Stop, start, pause, and resume a component of the current Siebel Server. • Stop, pause, and resume a task of the current component and Siebel Server.
Components	Review the status of any component running in the Siebel Enterprise.
Sessions	Review the status of any session running in the Siebel Enterprise.
Tasks	Review the status of any task running in the Siebel Enterprise.

Manageable Element Under Siebel Enterprise	Management Activity

Managing an Element in the Management Screen

To manage Siebel enterprise elements in the Management screen in the Siebel Management Console, perform the following general steps. The specific actions you can take depend on the navigation context, and are shown in the table in *Activities in the Management Screen*. This task is part of *Managing the Siebel Deployment*.

To manage an element in the Management screen

1. Run the Siebel Management Console, as described in *Starting the Siebel Management Console*.
2. Click Management in the navigation menu in the side panel.
3. For the current Siebel enterprise, use the navigation bar to navigate to one of the available options: Servers, Components, Sessions, or Tasks. Where appropriate, select a specific instance, such as a specific Siebel Server.
4. Perform one of the actions noted in the table in *Activities in the Management Screen*, whether for the Siebel Server or for a child element of either the Enterprise or the Siebel Server.

Accessibility for Siebel Management Console

Siebel Management Console supports enhanced accessibility.

Siebel Management Console Keyboard Shortcuts

The keyboard shortcuts (hotkeys) for regions within the Siebel Management Console user interface are shown in the following table.

Region	Keyboard Shortcut
Navigation menu	Alt+N
Toolbar or view overview level	Alt+O
List view	Alt+L
Details or form view	Alt+C
Navigation bar (breadcrumb)	Alt+M
Global search	Alt+G
Copyright	Alt+R

Region	Keyboard Shortcut

A user can also use various keyboard keys to move the focus through the user interface controls in Siebel Management Console, and to execute functions, as appropriate for the control that has the focus. Try these keys in various contexts to fully understand their actions. The tables that follow provide details about keyboard navigation options in some of these contexts.

For information about accessibility for Siebel CRM applications, see *Siebel Fundamentals Guide* .

Navigation Menu Keyboard Shortcuts

In the Navigation menu (accessible using Alt+N), the following keyboard shortcuts are available, as shown in the following table.

Keyboard Shortcut	Action
Forward Arrow	When focus is on a closed node, opens the node; focus does not move. When focus is on an open node, moves focus to the first child node. When focus is on an end node, does nothing.
Back Arrow	When focus is on an open node, closes the node. When focus is on a child node that is also either an end node or a closed node, moves focus to its parent node. When focus is on a root node that is also either an end node or a closed node, does nothing.
Down Arrow	Moves focus to the next node that is focusable without opening or closing a node.
Up Arrow	Moves focus to the previous node that is focusable without opening or closing a node.
End	Moves focus to the last node in the tree that is focusable without opening a node.
Home	Moves focus to the first node in the tree without opening or closing a node.
Enter	Selects the node. When focus is on a parent node, opens or closes the node.

View Overview Keyboard Shortcuts

In the toolbar or view overview section (accessible using Alt+O), the following keyboard shortcuts are available, as shown in the following table.

Keyboard Shortcut	Action
Forward Arrow	Moves focus between the view title and action buttons.

Keyboard Shortcut	Action
Back Arrow	
Enter or Space	Selects the action button with the focus.

List View Keyboard Shortcuts

In a list view (accessible using Alt+L), the following keyboard shortcuts are available, as shown in the following table.

Keyboard Shortcut	Action
Nonquery mode	
Forward Arrow	Moves focus to the cell after the current one (if it exists).
Back Arrow	Moves focus to the cell before the current one (if it exists).
Down Arrow	Moves focus to cell below the current one (if it exists).
Up Arrow	Moves focus to cell above the current one (if it exists).
Page Down	Views the next page (if it exists) and moves focus to form title.
Page Up	Views the previous page (if it exists) and moves focus to form title.
End	Views the last page (if not on last page) and moves focus to form title.
Home	View the first page (if not on first page) and moves focus to form title.
Enter	When focus is on the header, sorts by the column and moves focus to form title.
Tab / Shift+Tab	Moves focus to page status and other areas.
Query mode	
Tab	Moves focus to next focusable control.
Shift+Tab	Moves focus to previous focusable control.
Enter	When focus is on a query operator, opens the dropdown list. When focus is in a dropdown list, selects the item and moves focus back to query operator.

Keyboard Shortcut	Action
	When focus is on input, executes the query.
Escape	When focus is on query option dropdown list, closes it.
Down Arrow	When focus is in dropdown list, moves focus to next item.
Up Arrow	When focus is in dropdown list, moves focus to previous item.
Page Down	View next page (if it exists) and moves focus to form title.
Page Up	View previous page (if it exists) and moves focus to form title.
End	When focus is in dropdown list, moves focus to the last item.
Home	When focus is in dropdown list, moves focus to the first item.

Form View Keyboard Shortcuts

In a details or form view (accessible using Alt+C), the following keyboard shortcuts are available, as shown in the following table.

Keyboard Shortcut	Action
Tab	Moves focus to next item (session, control, error message, buttons, page navigation tabs).
Shift+Tab	Moves focus to previous item (session, control, error message, buttons, page navigation tabs).
Forward Arrow or Back Arrow	When focused on page navigation tabs, moves focus to the next or previous page. When focus is on a radio button, selects the next or previous button. When focus is on a list-box, selects the option.
Space or Enter	When focus is on section header (toggle), expands or collapses the section. When focus is on buttons (Add, Clone, Delete, Previous, Next, Submit, Cancel, Advanced), activates the button. When focus is on a checkbox, checks or unchecks the box. When focus is on a list-box, activates the dropdown list or confirms selected option.

Navigation Bar (Breadcrumb) Keyboard Shortcuts

In the breadcrumb or navigation bar (accessible using Alt+M), which is found in the Configuration screen, the following keyboard shortcuts are available, as shown in the following table.

Keyboard Shortcut	Action
Forward Arrow	When focus is in breadcrumb bar, moves focus to the next item, wrapping from the last to the first if necessary. When focus is in a dropdown menu and on a menu item that has a submenu, moves focus to its first item. When focus is in a dropdown menu and on a menu item that does not have a submenu, moves focus to the next item in the breadcrumb bar.
Back Arrow	When focus is in the breadcrumb bar, moves focus to the previous item, wrapping from the last to the first if necessary. When focus is in a submenu of an item in a dropdown menu, closes the submenu and returns focus to the parent menu item. When focus is in a submenu of a breadcrumb item, moves focus to the previous item in the breadcrumb bar.
Down Arrow	When focus is in the breadcrumb bar, moves focus to the first item in the dropdown menu. When focus is in a dropdown menu, moves focus to the next item, wrapping from the last to the first if necessary.
Up Arrow	When focus is in the breadcrumb bar, moves focus to the last item in the dropdown menu. When focus is in a dropdown menu, moves focus to the previous item, wrapping from the first to the first if necessary.
Space or Enter	When focus is in the breadcrumb bar, moves focus to the corresponding configuration view.
Tab or Shift+Tab	Moves focus to other elements of the page.

Global Search Keyboard Shortcuts

In the global search (accessible using Alt+G), the following keyboard shortcuts are available, as shown in the following table.

Keyboard Shortcut	Action
Down Arrow	When dropdown list is opened and focus is in input, moves focus to the first item in dropdown list. When dropdown list is opened and focus is inside dropdown list, moves focus to next item.
Up Arrow	When dropdown list is opened and focus is inside dropdown list, moves focus to previous item.

Keyboard Shortcut	Action
End	When dropdown list is opened and focus is inside dropdown list, moves focus to the last item.
Home	When dropdown list is opened and focus is inside dropdown list, moves focus to the first item.
Enter or Space	When dropdown list is opened and focus is inside dropdown list, navigates to the focused item.
Escape	When dropdown list is opened and focus is inside dropdown list, closes the dropdown list and moves focus to input.

Related Topics

About Configuring Siebel CRM

Starting the Siebel Management Console

Overview of Siebel Management Console

Supported Activities in Siebel Management Console

Configuring the Siebel Deployment

Managing the Siebel Deployment

5 Installing the Siebel Database on the RDBMS

Installing the Siebel Database on the RDBMS

This chapter describes running the Database Configuration Wizard to install the Siebel database and related tasks. This chapter is written for administrators who run the Database Configuration Wizard and for database administrators (DBAs) who perform related tasks on the relational database management system (RDBMS). This chapter includes the following topics:

- *About Database Updates for Siebel CRM*
- *About Installing the Siebel Database*
- *Creating Table Owner and Administrator Accounts*
- *Installing the Stored Procedures and User-Defined Functions on IBM DB2*
- *Starting the Siebel Database Configuration Wizard*
- *Installing the Siebel Database*
- *Activating License Keys*
- *Setting Up Your Environment to Support Global Time Zone*
- *Populating the Siebel File System*
- *Importing a Siebel Repository Manually into the Siebel Database*

Note: If your database is IBM DB2 for z/OS, then see *Implementing Siebel Business Applications on DB2 for z/OS* instead of this chapter. See also the relevant information in *Planning RDBMS Installation and Configuration*.

About Database Updates for Siebel CRM

Tasks related to the Siebel database vary, according to your installation case, corresponding to the roadmap topics identified in the following paragraphs.

- After performing a new installation of Siebel CRM (in a new deployment), you use the Database Configuration Wizard task to install a new Siebel database, as described in this chapter. This database contains schema changes, Siebel Repository content, and seed data for Siebel CRM functionality through the current release. The PostInstallDBSetup and RepositoryUpdate utilities, described later in this topic, do not apply. See also *Roadmap: Installing Siebel CRM for a New Deployment*.
- When you install Siebel CRM as an update for an existing deployment of Siebel CRM 17.x or later, the installer runs the Post Installation Database Update (PostInstallDBSetup) utility, as described later in this topic. This utility must run for each subsequent update release you install. You can also optionally run the RepositoryUpdate utility manually, as described later in this topic. See also *Roadmap: Installing Siebel CRM in an Update Installation Case (Existing Installation of Siebel CRM 25.x or Later)*.
- After performing a migration installation of Siebel CRM, you run Incremental Repository Merge (IRM) to apply the latest database updates. The PostInstallDBSetup and RepositoryUpdate utilities, described later in this topic, do not apply. See also *Roadmap: Installing Siebel CRM in a Migration Installation Case (Existing Installation of Siebel CRM 16.x or Earlier)*.

- After performing a new installation of Siebel CRM in a full upgrade installation case, you perform a full database upgrade to apply the latest database updates. The PostInstallDBSetup and RepositoryUpdate utilities, described later in this topic, do not apply. See also *Roadmap: Installing Siebel CRM in an Upgrade Case (Existing Database Requiring Full Upgrade)*.

About the PostInstallDBSetup Utility

The PostInstallDBSetup utility (also called Post Installation Database Update) runs whenever you run the Siebel CRM installer to update the installed Siebel CRM enterprise components. Where applicable, you run PostInstallDBSetup on every Siebel database (that is, both development and runtime repository environments). PostInstallDBSetup was first provided in Siebel CRM 20.3.

PostInstallDBSetup runs several processes to ensure that the customer database schema, manifest, and seed data is up to date for the current monthly update release. Some other utilities that were formerly required for customers to run are now included in the functionality of PostInstallDBSetup.

As noted, PostInstallDBSetup applies to update installations. It does not apply to installation cases where you are installing a new Siebel database or running Incremental Repository Merge.

The Siebel CRM installer screen for collecting database details for PostInstallDBSetup includes a menu with the following options, as appropriate for your use case:

- **Execute.** Collects the database details you provide and executes PostInstallDBSetup. Use this option in a typical update installation case. You must provide valid values to be passed to PostInstallDBSetup. If PostInstallDBSetup does not complete correctly or if you provide invalid values, then the utility exits. Where applicable, you must then run PostInstallDBSetup manually. Your Siebel CRM applications will not be in a supported state until PostInstallDBSetup has successfully been run after installation.
- **Defer-Generate DDL files only.** Collects the database details you provide and generates a SQL file containing Data Definition Language scripts for running PostInstallDBSetup manually later. The installer does not run PostInstallDBSetup. Use this option in any installation in which you plan to run PostInstallDBSetup manually after completing the installation. Your Siebel CRM applications will not be in a supported state until PostInstallDBSetup has successfully been run after installation.
- **Skip.** The installer does not run PostInstallDBSetup and any data you specify is ignored. Use this option in a deployment where you previously updated an installation for the same version in another location and ran PostInstallDBSetup. Also use this option in any other installation case in which PostInstallDBSetup is not used.

For more information about running PostInstallDBSetup manually, see *Siebel Database Upgrade Guide* .

About the RepositoryUpdate Utility

Optionally, after installing Siebel CRM as an update for an existing deployment of Siebel CRM 17.x or later and allowing PostInstallDBSetup to run, you can run the RepositoryUpdate utility manually on the development database to install new features since Siebel CRM 17.0 that require repository updates. This task does not apply in any other environments or if you do not require any of these new features. For more information about running this utility and about the new features that require repository updates, see *Siebel Database Upgrade Guide* and *Siebel CRM Update Guide and Release Notes* on My Oracle Support for your Siebel CRM release.

Related Topics

Installation-Related Changes in Siebel CRM

About Installing the Siebel Database

About Siebel CRM Releases

About Installing in Upgrade Environments

Related Books

Siebel Database Upgrade Guide

Siebel CRM Update Guide and Release Notes on My Oracle Support

About Installing the Siebel Database

> **CAUTION:** In an upgrade environment, or in another environment where you have an existing Siebel database, you generally do not perform any of the tasks mentioned in this chapter. For more information, see *About Installing in Upgrade Environments* and *Roadmap: Installing Siebel CRM in an Upgrade Case (Existing Database Requiring Full Upgrade)*.

The *Siebel database* on the RDBMS stores the data used by Siebel CRM. Siebel Server components (particularly Application Object Managers supporting Siebel Web Clients), Siebel Tools, and Siebel Developer Web Clients connect directly to the Siebel database and make changes in real time.

Siebel Mobile Web Clients download a subset of the server data to use locally. The mobile user periodically synchronizes the local database with the Siebel database through the Siebel Remote components on the Siebel Server. The synchronization process updates both the local database and the Siebel database.

The *Siebel Database Configuration Utilities* (also called the *Siebel Database Server*) refers to a set of files that you install on a Siebel Server computer, using the Siebel CRM installer. These files are accessed when you run the Database Configuration Wizard and the Siebel Upgrade Wizard in order to install and configure the Siebel database on the RDBMS.

In the Database Configuration Wizard, the main task that you perform for a new deployment is called *installing* the Siebel database. Additional tasks described in this chapter are sometimes referred to by using the general phrase *configuring* the Siebel database.

> **Note:** Database Configuration Utilities is no longer a selectable option for new installations. The `dbsrvr` directory is created for all new installations of Siebel Enterprise Components or migration installations of Siebel Enterprise Server from prior releases. For installation with Siebel Enterprise Components, the New Database Repository and Ancestor Database Repository selections are available as options, where the product media for these modules was included in the Siebel installation image.

For instructions on installing Database Configuration Utilities, Siebel Server, and other Siebel Enterprise Server modules, see *Siebel CRM Download and Installation*.

> **Note:** This chapter assumes that you have already created the database instance. It also assumes that you have already installed the Database Configuration Utilities on the same computer as a Siebel Server, and in the same top-level installation directory. You install the Database Configuration Utilities only once, typically with the first Siebel Server that you install.

For a new deployment, you must install a Siebel Server and the Database Configuration Utilities on the same computer as the Siebel Gateway. For more information, see *Requirements for Siebel Gateway Authentication*. For information about the tasks that you must perform in the RDBMS before you install the Database Configuration Utilities, see *Configuring the RDBMS*.

The Database Configuration Utilities installed software has no runtime role in managing database operations for users running Siebel CRM. Some of the tasks that you can perform with the Database Configuration Wizard are for upgrades only, and are described in *Siebel Database Upgrade Guide* .

> **Note:** After performing a new database installation, you must activate the license keys, as described in *Activating License Keys*. Then run the Siebel Management Console as described in this book.

This topic includes the following information:

- *About Installing a Database*
- *About Adding a Language to an Existing Database*

Related Topics

Siebel CRM Download and Installation

Configuring Siebel CRM Server Modules

About Database Updates for Siebel CRM

Activating License Keys

Configuring the RDBMS

Related Books

Siebel Database Upgrade Guide

About Installing a Database

When you choose Install Database from the Siebel Database Operation screen in the Database Configuration Wizard, the utility performs several tasks within the database instance that you have already created. This topic is part of *About Installing the Siebel Database*.

The Database Configuration Wizard is available on the computer on which you installed the Siebel Server and the Database Configuration Utilities.

Before installing the Siebel database, review all of the applicable requirements. The Install Database operation does the following:

1. Creates the Siebel schema. Tables and indexes are created in their respective specified tablespaces.
2. Inserts Siebel seed data specific to your database. Seed data is inserted for the primary language only. This task also installs views, packages, and procedures for your database.
3. Imports the Siebel Repository. The Repository is imported for the primary language only.
4. Sets system preferences.

About Adding a Language to an Existing Database

Adding a language to an existing Siebel database installs seed data to support that language. After the database is initially installed with the primary language, each additional language must be installed separately using an add-language operation. This topic is part of *About Installing the Siebel Database*.

To add a new language to previously installed Siebel CRM, you must have installed its Siebel language on the installed Siebel Server, using the Siebel CRM installer. See also *Installing and Deploying Siebel CRM with Multiple Languages*.

After adding a language, you must import the Siebel Repository for this language. For more information, see *Importing a Siebel Repository Manually into the Siebel Database*.

After you have added a language to the Siebel database, your seed data is multilingual. You must enable the multilingual list of values (MLOV) capability for Siebel CRM, and you must enable individual LOVs associated with the language. For more information, see:

- *Configuring Siebel Business Applications*
- *Siebel Global Deployment Guide*

Creating Table Owner and Administrator Accounts

Create the table owner and administrator accounts according to the guidelines provided for each database platform identified in this topic. These accounts are created by updating and running the grantusr.sql script. Record all of the relevant data, such as the table owner name and password, in the worksheets provided in *Siebel Deployment Planning Worksheet*.

> **Note:** For each applicable database platform, if you are planning to use Siebel Marketing, then also grant DROP TABLE, DROP INDEX, CREATE TABLE, and CREATE INDEX rights at the database level within the OLTP schema to the table owner or the database user ID used for Siebel Marketing. For more information, see *Siebel Marketing Installation and Administration Guide* .

> **CAUTION:** It is strongly recommended not to change the name of the Siebel administrator account, SADMIN. This account must be created so that you can log in to Siebel applications as the Siebel administrator. For information about changing the password for this account, see *Siebel Security Guide* . The password for the SADMIN account must not exceed 18 characters.

> **Note:** Before you execute grantusr.sql, confirm that this script will create all of the users that must be created before you install the Siebel database. Optionally, you can also modify the script to define the anonymous users that you will specify when configuring the Siebel Application Interface. For more information, see *Requirements for Configuring Anonymous Users for the Siebel Application Interface*.

This topic includes the following information:

- *Running the grantusr.sql Script on an Oracle Database*
- *Running the grantusr.sql Script on IBM DB2*

- *Running the grantusr.sql Script on Microsoft SQL Server*
- *Running the grantusr.sql scripts on Azure SQL*

Running the grantusr.sql Script on an Oracle Database

Before you install the Siebel database, your DBA must review, modify, and execute the grantusr.sql script. This task is part of *Creating Table Owner and Administrator Accounts*.

The administrator executes grantusr.sql against the database instance to create the Siebel table owner account, the Siebel administrator account (default: SADMIN), the LDAPUSER account, and the role SSE_ROLE and to grant them appropriate privileges. The script also creates the default tablespace. The grantusr.sql script is located in the `DBSRVR_ROOT\ORACLE` subdirectory.

After executing grantusr.sql, the DBA must manually create all of the additional tablespaces (for data or indexes) that you require in your Siebel database. To do this, use commands similar to the existing ALTER USER command defined in grantusr.sql for creating the default tablespace. Use commands similar to the following:

```
ALTER USER TABLE_OWNER quota unlimited on tablespace_name;
```

where:

- TABLE_OWNER is the name of the Siebel table owner account.
- tablespace_name is the name that you are using for each particular tablespace. If it is necessary, then the quota value can be adjusted to an appropriate value for each tablespace.

The grantusr.sql script performs the following functions:

- Creates the role SSE_ROLE and grants CREATE SESSION privilege to this role.
- Creates the Siebel table owner account and grants other appropriate privileges to this account.
- Creates the accounts SADMIN (the Siebel administrator) and LDAPUSER and grants the role SSE_ROLE to them.
- Creates the default tablespace.

Passwords are prompted for by the script, rather than defined within the script.

The length and allowable characters for the user name and password depend on the rules of your underlying RDBMS platform. For more information, see Oracle Database documentation.

To run the grantusr.sql script on an Oracle Database

1. Execute the grantusr.sql script from SQL*Plus, using an account with DBA privileges, and using the following command:

   ```
   @c:\SIEBEL_ROOT\dbsrvr\oracle\grantusr.sql
   ```

   > **Note:** You must specify the full path to the grantusr.sql file.

2. Enter the tablespace name listed in *Siebel Deployment Planning Worksheet*.

Running the grantusr.sql Script on IBM DB2

Before you install the Siebel database, your DBA must manually create the Siebel table owner account, the Siebel administrator account (default: SADMIN), and the SSE_ROLE group. The DBA must then add the Siebel administrator

account to the SSE_ROLE group at the operating system level. This task is part of *Creating Table Owner and Administrator Accounts*.

Before you install the Siebel database, execute the grantusr.sql script against your database server to grant the appropriate privileges to these users. The grantusr.sql script is located in the `DBSRVR_ROOT\DB2UDB` subdirectory.

Your DBA must review and execute this script, which performs the following functions:

- Grants DBA administration (DBADM) privileges to the table owner account.
- Grants CONNECT privileges to the SSE_ROLE group.

You cannot create the LDAPUSER account by running grantusr.sql. This account must belong to the SSE_ROLE group and must be created by the DBA or the Windows network administrator, as appropriate. For more information about authentication using the LDAP security adapter, see *Siebel Security Guide* .

This topic is part of *Creating Table Owner and Administrator Accounts*.

To run the grantusr.sql script on IBM DB2

1. Execute the grantusr.sql script from a DB2 Command Window, using an account with DBA privileges.

   The usual DB2 System Administration account is called `db2admin` for this procedure.

   > **CAUTION:** Use the DB2 Command Window, not the DB2 command line processor (CLP), to enter these commands, because the DB2 command line processor window uses different syntax. The commands in this procedure work only if you issue them in a DB2 Command Window.

2. Enter the following commands:

   ```
   db2 connect to DB2database_alias user instance_owner_username using password
   db2 -vf SIEBEL_ROOT\DBSRVR\DB2UDB\grantusr.sql
   ```

   where:

   - DB2database_alias is the DB2 alias that you use
   - instance_owner_username is the login ID of the database instance owner
   - password is the password for the database instance owner
   - SIEBEL_ROOT is the full path to the Siebel root directory

   The script prompts you for the default tablespace in which your Siebel objects are to be created.
3. Enter the tablespace name that you recorded in the copy that you made of *Siebel Deployment Planning Worksheet* and exit the script.

The length and allowable characters of the account names and passwords depend on the rules of your underlying RDBMS platform. For more information, see IBM DB2 documentation.

Before you install the Siebel database, see *Installing the Stored Procedures and User-Defined Functions on IBM DB2*.

Running the grantusr.sql Script on Microsoft SQL Server

Before you install the Siebel database, your DBA must review, modify, and execute the grantusr.sql script. This task is part of *Creating Table Owner and Administrator Accounts*.

The administrator executes grantusr.sql against the database to set up minimum security. The grantusr.sql script is located in the `DBSRVR_ROOT\MSSQL` subdirectory.

The grantusr.sql script performs the following functions:

- Creates database accounts (logins) for the SADMIN (Siebel administrator) account, the table owner account, and the LDAPUSER account.
- Creates users for each of these accounts, except the table owner account, which will be turned into a dbo by the stored procedure sp_changedbowner.
- Grants SSE_ROLE to each account, except dbo.

To change the account name or the database name, edit grantusr.sql. The length and allowable characters of the account names and passwords depend on the rules of your underlying RDBMS platform. For more information, see Microsoft SQL Server documentation.

To run the grantusr.sql script on Microsoft SQL Server

1. Open grantusr.sql in Microsoft Query Analyzer.
2. Execute the script.

Running the grantusr.sql scripts on Azure SQL

This topic explains how to run the grantusr_azure_master.sql and grantusr_azure_siebeldb.sql scripts to set up necessary logins and users for Siebel CRM on Azure SQL.

Before you install the Siebel database, the Database Administator (DBA) must review, modify, and execute the **grantusr_azure_master.sql** and **grantusr_azure_siebeldb.sql** scripts.

This task is part of *Creating Table Owner and Administrator Accounts*.

> **Note:** For Azure SQL Databases, you must split your traditional grantusr.sql script into two separate scripts, but these scripts do the same fundamental steps as are done against Microsoft SQL Server (On Premises) databases.

Both the scripts are located in the DBSRVR_ROOT\MSSQL subdirectory.

> **Note:** For purposes of these instructions, the following assumptions are made:
> - Your database owner is "SIEBEL"
> - Your Siebel database is named as "siebeldb"
>
> You must adjust the scripts if either of these is not the case.

As an administrator execute these scripts against the database, to set up minimum database logins and users.

1. **Prepare and run the grantusr_azure_master.sql script:**

   The grantusr_azure_master.sql script creates database logins for the SADMIN (Siebel administrator) account, the table owner (SIEBEL) account, various GUEST* accounts, and the LDAPUSER account.

   a. Specify the desired passwords for these users in the following lines at the top of the script:

      For example, for the SIEBEL user, update the following line:

`INSERT INTO #Logins VALUES (1, N'SIEBEL', '');` to `INSERT INTO #Logins VALUES (1, N'SIEBEL', '********');` Where ******** is your desired password.

b. You may also optionally add more logins if you plan to use Database Authentication in this environment, but Siebel CRM Production environments should not use Database Authentication.

c. After editing the script, execute it against the master db.

> **Note:** The script performs the following validations:
> - The password cannot be blank.
> - The password cannot be the same as the login name.
> - If the login already exists, it will not be recreated.
> - Any database password complexity or other requirements will be enforced by the database, not the script.

2. **Prepare and run the grantusr_azure.siebeldb.sql script**

The grantusr_azure_siebeldb.sql script creates users in the siebeldb.

> **Note:** For an explanation of the difference between logins and users, refer to the Microsoft documentation.

a. Before running the script, the DBA must ensure that the following variables at the top of the script are set:
- --@Role specifies the database role that will be given to all the other users to provide access to the Siebel data. This is typically **"SSE_ROLE"**.
- --@TableOwner specifies the database owner (dbo) for the siebeldb. This is typically **SIEBEL**.
- --@systemAdmin is used temporarily during the process of changing the dbo to the @TableOwner. This can be almost any user (other than @TableOwner), but for simplicity, just use the login of the system user that is executing the script.

b. After editing the script, execute it against the siebeldb.

Installing the Stored Procedures and User-Defined Functions on IBM DB2

On IBM DB2, after installing the Database Configuration Utilities, you must copy stored procedures and user-defined functions (UDFs) from the Database Configuration Utilities installation to the DB2 database server.

Any method that transfers the necessary files to the correct location on the database server is acceptable. To copy the stored procedure code, complete the procedure that follows. Use the steps appropriate to the operating systems for the Database Configuration Utilities and the RDBMS. For information about how to perform basic IBM DB2 tasks, see the *Quick Beginnings* guide from IBM.

To copy and install the stored procedures and user-defined functions

1. Log on to the Siebel Server computer on which you installed the Database Configuration Utilities.
2. Navigate to the following subdirectory in the Database Configuration Utilities installation directory:

```
DBSRVR_ROOT\DB2UDB\SIEBPROC\DBSRVR_OS
```

where:

- `DBSRVR_ROOT` is the Database Configuration Utilities module subdirectory of your Siebel CRM installation directory (`SIEBEL_ROOT\dbsrvr`).
- DBSRVR_OS is the string corresponding to the operating system your database server (RDBMS) runs on, such as `WIN32`.

3. Copy the file siebproc.dll (Windows) or siebproc (UNIX) to the `FUNCTION` subdirectory within the DB2 instance directory on the RDBMS computer where DB2 is installed.

   For example, on Windows, this location might be `C:\SQLLIB\FUNCTION` or `C:\Program Files\SQLLIB\FUNCTION`. On AIX, this location might be `$INST_HOME/sqllib/function`.

   (UNIX) Permissions for siebproc must be -rw-r---- (read-write for the owner, read for the group, and neither read nor write for all others). The owner of the file must be the instance owner, and the group must be the group to which the instance owner belongs.

   For a 64-bit DB2 database, do one of the following instead:

   - For a 64-bit DB2 database running on Itanium architecture, copy the file siebproc64.dll to the `SQLLIB\FUNCTION` directory, and rename it to siebproc.dll (Windows) or siebproc (UNIX).
   - For a 64-bit DB2 database running on X64 architecture, copy the file siebprocEMT64.dll to the `SQLLIB\FUNCTION` directory, and rename it to siebproc.dll (Windows) or siebproc (UNIX).
4. Proceed to *Installing the Siebel Database*.

Starting the Siebel Database Configuration Wizard

Use the following procedure to start the Siebel Database Configuration Wizard.

- For Microsoft Windows, you start the wizard by
  - Using a Windows shortcut, as described in *Starting the Siebel Database Configuration Wizard Using a Windows Shortcut*.
  - From the command line, as described in *Starting the Siebel Database Configuration Wizard at the Command Line on Microsoft Windows*.
- For UNIX, you start the wizard from the command line, as described in *Starting the Siebel Database Configuration Wizard on UNIX*.

Starting the Siebel Database Configuration Wizard Using a Windows Shortcut

Use the following procedure to start the Siebel Database Configuration Wizard by using a Windows shortcut. This task is part of *Starting the Siebel Database Configuration Wizard*.

To start the Siebel Database Configuration Wizard using a Windows shortcut

1. On a computer where you installed Siebel Enterprise Server software, choose Start, All Programs, Siebel Enterprise Server 22.x, and then Database Server Configuration. Go to the next step.
2. Respond to each prompt in the Database Configuration Wizard, then click Next to go to the next screen, or to complete the configuration.

   See *Installing the Siebel Database*.

   Note: You can click Cancel to exit the Database Configuration Wizard before you complete the configuration.

   After you specify or confirm all of your selections, the summary screen appears.
3. In the summary screen, do one of the following:
   - Click Next again to complete the configuration. Go to the next step.
   - Optionally, save a response file in this screen, to use in an unattended configuration, then click Cancel to exit the Database Configuration Wizard without completing configuration.
4. After configuration is complete, click Exit to exit the Database Configuration Wizard.

   To help you to verify that the configuration completed successfully, the log file is referenced at the end of the configuration task.

Starting the Siebel Database Configuration Wizard at the Command Line on Microsoft Windows

Use the following procedure to start the Siebel Database Configuration Wizard at the command line on Windows. This task is part of *Starting the Siebel Database Configuration Wizard*.

To start the Siebel Database Configuration Wizard at the command line

1. Open a DOS command window.
2. Navigate to the `config` subdirectory within your `SIEBEL_ROOT` directory.

   For example, navigate to a directory like the following, on the Siebel Server computer: `C:\Siebel\config`.

   To start the Database Configuration Wizard to configure in GUI mode, go to the next step. Use the same commands if you are generating a response file for unattended configuration. To perform an unattended configuration, go to Step 4.
3. Enter a command similar to the following to start the Database Configuration Wizard:

   ```
   install_path\config\config.bat -mode dbsrvr
   ```

   In this command, install_path is the installation path for the installed Siebel CRM module, such as in the following example command:

   ```
   C:\Siebel\config\config.bat -mode dbsrvr
   ```

   Additional command-line options that you can use are:

- **-skipValidation.** Specifies that no validation should be performed of user input. You can use this flag when saving a response file for an unattended configuration, or when previewing wizard screens.
- **-verbose.** Specifies that more details should be included in configuration log files.

For an unattended configuration, go to the next step. Otherwise, go to Step 5.

4. For an unattended configuration, you save and later execute a response file for configuring the Siebel database. Do one of the following:

   - If you are generating a response file, then go to Step 5 and continue the configuration process. In the summary screen, choose the option to save a configuration file, then cancel the Database Configuration Wizard. Then, to perform the unattended configuration of the same module, restart the wizard as described in the next paragraph.
   - If you are executing a response file that you saved in a previous configuration session, then start the wizard using the `-responseFile` option, specifying the name of the response file to execute. Use a command similar to the following:

     ```
     install_path\config\config.bat -mode dbsrvr -responseFile file_name
     ```

     where:

     - install_path is the installation path for the installed Siebel CRM module (see example in Step 3)
     - file_name is the name of the response file that you are executing

     > **Note:** For a command like this, the response file would have been saved in a previous configuration session. In unattended mode, wizard screens that would normally display are not displayed, because the applicable data is already stored in the response file.

     If you are generating a response file, then go to the next step. If you are executing a response file, then the Database Configuration Wizard runs in unattended mode and completes configuration.

5. Respond to each prompt in the Database Configuration Wizard, then click Next to go to the next screen or to complete the configuration.

   See *Installing the Siebel Database*.

   > **Note:** You can click Cancel to exit the Database Configuration Wizard before you complete the configuration.

   After you specify or confirm all of your selections, the summary screen appears.

6. In the summary screen, do one of the following:

   - Click Next again to complete the configuration. Go to the next step.
   - Optionally, save a response file in this screen, to use in an unattended configuration, then click Cancel to exit the Database Configuration Wizard without completing configuration.

7. After configuration is complete, click Exit to exit the Database Configuration Wizard.

   To help you to verify that the configuration completed successfully, the log file is referenced at the end of the configuration task.

Starting the Siebel Database Configuration Wizard on UNIX

Use the following procedure to start the Siebel Database Configuration Wizard on UNIX. You start the wizard from the command line. This task is part of *Starting the Siebel Database Configuration Wizard*.

As part of the following procedure, you must enter the necessary command to source one of the following environment setup files, as appropriate, to configure or validate the environment.

- **dbenv.csh or dbenv.sh.** Before you run the Database Configuration Wizard, you must source either dbenv.csh (for C shell) or dbenv.sh (for Bourne or Korn shell). Before you can source one of these files, you must create them by running the script CreateDbSrvrEnvScript. These environment files are created in the `$SIEBSRVR_ROOT` installation directory. For more information, see *Installing the Siebel Database*.

To start the Siebel Database Configuration Wizard at the command line

1. Open a shell window.
2. Source an environment setup file, as appropriate:

   Before you run the Siebel Database Configuration Wizard, source either dbenv.csh (for C shell) or dbenv.sh (for Bourne or Korn shell). For more information, see *Installing the Siebel Database*.
3. Navigate to the `config` subdirectory within your `$SIEBEL_ROOT` directory.

   For example, navigate to a directory like the following, on the Siebel Server computer: `/Siebel/config`.

   To start the Database Configuration Wizard to configure in GUI mode, go to the next step. Use the same commands if you are generating a response file for unattended configuration. To perform an unattended configuration, go to Step 5.
4. Enter a command similar to the following to start the Database Configuration Wizard:

   ```
   install_path/config/config -mode dbsrvr
   ```

   In this command, install_path is the installation path for the installed Siebel CRM module, such as in the following example command:

   ```
   /Siebel/config/config -mode dbsrvr
   ```

   Additional command-line options that you can use are:

   - **-skipValidation.** Specifies that no validation should be performed of user input. You can use this flag when saving a response file for an unattended configuration, or when previewing wizard screens.
   - **-verbose.** Specifies that more details should be included in configuration log files.

     > **Note:** Using the -verbose option is recommended in order to log information about certain issues that might affect your configuration tasks. For example, various 32-bit libraries are required for successful configuration. When you use the -verbose option to generate a finer level of granularity in log messages, the Database Configuration Wizard might log, for example, that a particular required 32-bit library is missing.

   For an unattended configuration, go to the next step. Otherwise, go to Step 6.

5. For an unattended configuration, you save and later execute a response file for configuring the Siebel database. Do one of the following:

   - If you are generating a response file, then go to Step 6 and continue the configuration process. In the summary screen, choose the option to save a configuration file, then cancel the Database Configuration Wizard. Then, to perform the unattended configuration of the same module, restart the wizard as described in the next paragraph.
   - If you are executing a response file that you saved in a previous configuration session, then start the wizard using the `-responseFile` option, specifying the name of the response file to execute. Use a command similar to the following:

     ```
     install_path/config/config -mode dbsrvr -responseFile file_name
     ```

     where:

     - install_path is the installation path for the installed Siebel CRM module (see example in Step 4)
     - file_name is the name of the response file that you are executing

   > **Note:** For a command like this, the response file would have been saved in a previous configuration session. In unattended mode, wizard screens that would normally display are not displayed, because the applicable data is already stored in the response file.

   If you are generating a response file, then go to the next step. If you are executing a response file, then the Database Configuration Wizard runs in unattended mode and completes configuration.
6. Respond to each prompt in the Database Configuration Wizard, then click Next to go to the next screen or to complete the configuration.

   See *Installing the Siebel Database*.

   > **Note:** You can click Cancel to exit the Database Configuration Wizard before you complete the configuration.

   After you specify or confirm all of your selections, the summary screen appears.
7. In the summary screen, do one of the following:

   - Click Next again to complete the configuration. Go to the next step.
   - Optionally, save a response file in this screen, to use in an unattended configuration, then click Cancel to exit the Database Configuration Wizard without completing configuration.
8. After configuration is complete, click Exit to exit the Database Configuration Wizard.

   To help you to verify that the configuration completed successfully, the log file is referenced at the end of the configuration task.

Installing the Siebel Database

This topic describes how to install the Siebel database on the RDBMS. This task is a step in *Roadmap: Installing Siebel CRM for a New Deployment*. (Also perform all of the other applicable tasks in this chapter.)

Note: Installing a new database creates an ODBC data source name that allows you to use database authentication when you run Siebel Management Console, as described in *Configuring Siebel CRM Server Modules*.

This topic includes the following information:

- *Installing the Siebel Database on Microsoft Windows*
- *Installing the Siebel Database on UNIX*

Related Topics

About Installing the Siebel Database

Verifying Installation for the Siebel Database

Troubleshooting Installation for the Siebel Database

Installing the Siebel Database on Microsoft Windows

This topic describes how to install the Siebel database on the RDBMS, on Microsoft Windows. This task is part of *Installing the Siebel Database*.

To install Siebel database components on Microsoft Windows

1. Review *About Installing the Siebel Database* and *Starting the Siebel Database Configuration Wizard*.
2. Start the Database Configuration Wizard, as described in *Starting the Siebel Database Configuration Wizard*. Choose Start, All Programs, Siebel Enterprise Server 22.x, and then Database Server Configuration.
3. Specify the following, and click Next:

   Siebel Server Directory. Specify the existing installation location of Siebel Server. This location is also referred to as the `SIEBSRVR_ROOT` directory. For example, this directory might be `c:\Siebel\siebsrvr`.

   Siebel Database Server Directory. Specify the existing installation location of Siebel Database Configuration Utilities. This location is also referred to as the `DBSRVR_ROOT` directory. For example, this directory might be `c:\Siebel\dbsrvr`.
4. On the RDBMS Platform screen, select one of the following platforms, and click Next:
   - IBM DB2 UDB for Linux UNIX Windows
   - IBM DB2 UDB for z/OS
   - Microsoft SQL Server (Select this for Azure SQL DB Servers)
   - Oracle Database Enterprise Edition
5. On the Siebel Database Operation screen, select Install Database from the list, and click Next.
6. On the Select Installation Operation screen, select Install Siebel Database to install a new Siebel database, or select Add a Language to an Existing Siebel Database, and then click Next.

   The Install Siebel Database option creates the Siebel schema and inserts seed data and Siebel Repository data for the primary language at the end of the Configuration Wizard session.

   If you instead selected Add a Language to an Existing Siebel Database, then go to Step 15. You must have included the language when you installed the Siebel Server and Database Configuration Utilities. For more information, see *Installing and Deploying Siebel CRM with Multiple Languages*.

7. On the next screen, confirm that you want to install a new Siebel database, or indicate that you already have a Siebel database.
8. On the Siebel User/Role Creation screen, confirm that you (or a database administrator) have already run the grantusr*.sql script(s) to set up the table owner and administrator accounts.

   > **Note:** The table owner and administrator accounts must be in place or you will not be able to complete the installation of the tables, indexes, and seed data in the Siebel database. For more information, see *Creating Table Owner and Administrator Accounts*.

   - If you already ran grantusr.sql, then select the corresponding option, and click Next.
   - If you did not already run grantusr.sql, then exit the Database Configuration Wizard, and run the script now. When the script has finished executing, restart the Database Configuration Wizard.
9. (Oracle Database only) Specify the following, and click Next:

   Siebel Database Table Owner. Type the name of the Siebel database table owner or the account that will own the Siebel objects.

   Oracle SQLNet Connect String. Provide the name of the Oracle schema qualifier or table owner name and the SQLNet connect string for Oracle Database.
10. (Microsoft SQL/Azure SQL DB Servers only) Specify the following, and click Next:

    Microsoft SQL/Azure SQL Server Host Name. Provide the host name of the computer running Microsoft SQL/Azure SQL Server. This value must not be left empty.

    Microsoft SQL/Azure SQL Server Database Name. Provide the name of the Microsoft SQL/Azure SQL Server database.
11. (IBM DB2 or IBM DB2 for z/OS only) Specify the following, and click Next:

    DB2 Instance Path. Provide the IBM DB2 client installation path, such as `/home/sblqa1`, where `sblqa1` contains the `sqllib` subdirectory.
12. (IBM DB2 only) Specify the following, and click Next:

    Siebel Database Table Owner. Type the name of the Siebel database table owner or the account that will own the Siebel objects.

    IBM DB2 Database Alias. Provide the name of the IBM DB2 UDB schema qualifier or table owner name and the database alias for IBM DB2.
13. (IBM DB2 for z/OS only) Specify the following, and click Next:

    Siebel Database Table Owner. Type the name of the Siebel database table owner or the account that will own the Siebel objects.

    IBM DB2 Database Alias. Provide the name of the IBM DB2 UDB schema qualifier or table owner name and the database alias for IBM DB2.

    Current IBM DB2 SQL ID / Group Name. Provide the name of the IBM DB2 UDB SQL ID or group name.

14. On the Database Encoding screen, identify the appropriate database encoding method, and click Next:

    UNICODE Database

    Non-UNICODE Database

    > **CAUTION:** Choose the correct option for your database to prevent installing the wrong data types. The database cannot create Unicode data types on a non-Unicode page setting, so check this setting carefully before choosing this option.

    > **Note:** Before you select languages in the next two steps, review the information about code pages in *Planning RDBMS Installation and Configuration*.

15. On the Base Language screen, specify which language serves as the primary language (base language) for the Siebel database.

    > **Note:** If you installed a single Siebel language on the Siebel Server where you also installed Database Configuration Utilities, then that language is assumed for a new database installation. Neither the Base Language nor the Language Selection screen appears. Go to Step 17.

    The languages listed in this screen are those for which you previously installed Siebel languages on the Siebel Server where you also installed Database Configuration Utilities. Do one of the following, and click Next:

    - If you are installing a new Siebel database, then specify the language that will be the primary language. Seed data and Siebel Repository data will be installed for this language. Go to Step 17. (The Language Selection screen does not appear in this case.)
    - If you are adding a language to an existing Siebel database, then specify the language that you already installed as the primary language. Go to the next step to specify the additional language to install, from the Language Selection screen.

16. (Add Language case only) On the Language Selection screen, select the language that you want to install in the Siebel database, for an Add Language operation.

    > **Note:** For a new database install case, this screen does not appear. Go to the next step.

    The languages listed in this screen are those for which you previously installed Siebel languages on the Siebel Server where you also installed Database Configuration Utilities.

    Select the language that you want to add. Seed data will be installed for this language. You must import the Siebel Repository for this language separately (using the Add Language to an Existing Repository task), as described in *Importing a Siebel Repository Manually into the Siebel Database*.

    Some of the remaining screens do not apply for an add-language operation. Go to the next applicable step.

17. Specify the following, and click Next:

    Database User Name. Type the user name of the Siebel administrator; for example, `sadmin`.

    Database Password. Type the password for the Siebel administrator.

    Database Password (confirm). Retype the password to confirm it.

18. Specify the following, and click Next:

    Database Table Owner. Type the name of the Siebel database table owner or the account that will own the Siebel objects.

Database Table Owner Password. Type the password for the Siebel database table owner.

Database Table Owner Password (confirm). Retype the password to confirm it.

19. (Oracle Database only) On the Register External Oracle DB ODBC Driver screen, specify this option if you will use the Siebel Connector for Oracle Applications to exchange data with your back office system, and your back office system stores data in an Oracle Database.
20. (IBM DB2 only) Specify the following, and click Next:

    Index Table Space Name. Type the name for your index tablespace; for example, `SBL_INDX`. (Tablespace names must not contain spaces; underscores are allowed.)

    4K Table Space Name. Type the name for your 4-KB tablespace; for example, `TBS_4K`. (Tablespace names must not contain spaces; underscores are allowed.)

    16K Table Space Name. Type the name for your 16-KB tablespace; for example, `TBS_16K`. (Tablespace names must not contain spaces; underscores are allowed.)

    32K Table Space Name. Type the name for your 32-KB tablespace; for example, `TBS_32K`. (Tablespace names must not contain spaces; underscores are allowed.)
21. (Oracle Database only) Specify the following, and click Next:

    Index Table Space Name. Type the name for your index tablespace; for example, `INDEX01`. (Tablespace names must not contain spaces; underscores are allowed.)

    Table Space Name. Type the name for your tablespace; for example, `DATA01`. (Tablespace names must not contain spaces; underscores are allowed.)
22. (Microsoft SQL/Azure SQL DB Servers only) On the Repository Selection screen, specify if you are using binary sort order or case-sensitive sort order for the Siebel Repository, and click Next.
23. (Oracle Database and Microsoft SQL/Azure SQL Server only) On the Siebel License Key Option screen, specify that you will provide the license key data later, and click Next.

    The license keys for Siebel CRM base applications must be activated using the License Key Activation utility, as described in *Activating License Keys*.
24. (Oracle Database only) On the Oracle Parallel Index screen, specify the statement describing your database environment, and click Next:

    - Does not use the Oracle Parallel Indexing option
    - Uses the Oracle Parallel Indexing option

    If you have a single-CPU environment, then choose the first option. If you have a multiple-CPU environment, then choose the first or second option, depending on whether you are running your Oracle Database with parallel indexing on.
25. Specify the following, and click Next:

    Security Group ID / Grantee. Type the name for your security group. Specify `SSE_ROLE`.

26. On the Log Output Directory screen, accept the default log directory or enter a new directory name, and click Next.

    By default, logging occurs in the `SIEBSRVR_ROOT\log\operation\output` directory. In this path, operation corresponds to the operation that you are performing, such as *install* for an Install Database operation or *install_lang* for an Add Language operation.

    Click Help to view the following message, which asks if you are ready to apply configuration changes to the Siebel database now, using the Siebel Upgrade Wizard, or will do so later:

    ```
    Configuration is complete: your output will be saved under $SiebelRoot/siebsrvr/
    bin/master_<process>.ucf. Would you like to deploy the process you configured to
    the database now or later?
    ```

    > **Note:** Check the *Siebel CRM Update Guide and Release Notes* on My Oracle Support for the current release. If this document describes any tasks that you must perform to modify the generated SQL files, then modify the generated SQL files as necessary before you execute the SQL in the Siebel Upgrade Wizard.

27. Choose one of the following, and click Next:

    - **Yes apply configuration changes now.** Choose this option if you want the generated SQL files to be executed in the Siebel Upgrade Wizard.
    - **No I will apply configuration changes later.** Choose this option if you do *not* want the generated SQL files to be executed in the Siebel Upgrade Wizard. In this case, you must run the Upgrade Wizard manually at a later time in order to execute the generated SQL files.

    If you choose to apply configuration changes later, then the command line to apply the configuration later is:

    ```
    SIEBEL_ROOT\siebsrvr\bin\siebupg.exe /m master_operation.ucf
    ```

    For the purposes of this procedure, operation is either *install* or *install_lang*.

28. Review the values that you entered on the previous screens in the Database Configuration Wizard:

    - If you have to change any values, then click Back and Next to access the screens on which to change entries, and then to return to this screen.
    - Alternatively, you can click Cancel to cancel the Database Configuration Wizard session, then restart the wizard and enter different values.
    - When you have verified your input, click Next. Doing so generates SQL files that you will then apply to the database. When all of the SQL files have been generated, a message appears, stating `The configuration changes were applied successfully`. Click OK.

    If you chose No in Step 27, then the Siebel Upgrade Wizard does not appear. You must run it manually, after reviewing the generated SQL files.

    If you chose Yes in Step 27, then the Siebel Upgrade Wizard appears.

    > **Note:** When you run the Siebel Upgrade Wizard, if a program or system error occurs, then you can rerun the wizard, starting at the point at which the wizard failed. For more information, see *Siebel Database Upgrade Guide* .

    When the Siebel Upgrade Wizard finishes running, the Siebel Upgrade Wizard Complete screen appears. You have now finished installing the Siebel database components.

29. After the Siebel Upgrade Wizard has finished running, click Exit to exit the Database Configuration Wizard.

Installing the Siebel Database on UNIX

This topic describes how to install the Siebel database on the RDBMS, on UNIX operating systems. This task is part of *Installing the Siebel Database*.

To install Siebel database components on UNIX operating systems

1. Review *About Installing the Siebel Database* and *Starting the Siebel Database Configuration Wizard*.
2. Before you run the Database Configuration Wizard for the first time, run the CreateDbSrvrEnvScript script to generate the environment setup files dbenv.csh and dbenv.sh.

   CreateDbSrvrEnvScript is located in the directory `$SIEBSRVR_ROOT/install_script/install`. Run CreateDbSrvrEnvScript from this location by using the following command:

   ```
   ./CreateDbSrvrEnvScript Siebel_Root_Parent_Directory
    LANG
    DB_Platform
   ```

   where:

   - Siebel_Root_Parent_Directory is the top-level Siebel Enterprise Server installation directory.
   - LANG is the three-letter code for the primary language for the Siebel database that you will install; for example, ITA for Italian.
   - DB_Platform is Oracle, Db2Udb, or Db2.390.

   For example:

   ```
   ./CreateDbSrvrEnvScript /siebel ITA Oracle
   ```

   The dbenv.csh and dbenv.sh environment setup files are created in the Siebel Server installation directory.
3. Open a shell window.
4. Navigate to the `$SIEBSRVR_ROOT` directory and source the environment file by using one of the following commands, depending on the type of shell that you use. Perform this step each time that you run the Database

Configuration Wizard. You must have created this file first, by using the script CreateDbSrvrEnvScript, as described in Step 2.

C shell

```
source dbenv.csh
```

Bourne or Korn shell

```
. ./dbenv.sh
```

> **Note:** Make sure that there is a space between the initial period and `./dbenv.sh`.

In this path, `$SIEBSRVR_ROOT` is the location of the Siebel Server root directory. For example, the directory might be `/export/home/Siebel/siebsrvr`.

5. Verify whether the values for SIEBEL_ROOT and LANGUAGE are correctly set or are empty by entering:

```
env
```

If the variables are incorrect or missing, then reset them using one of the following commands, as appropriate to the shell that you use.

For example, the SIEBEL_ROOT value might resemble `/export/home/Siebel/siebsrvr`, while the value for LANGUAGE might be ENU, FRA, JPN, CHS, or another code representing the language in which you are running the Database Configuration Wizard.

C shell

```
setenv SIEBEL_ROOT New_Value

setenv LANGUAGE New_Value
```

Bourne or Korn shell

```
export SIEBEL_ROOT=New_Value

export LANGUAGE=New_Value
```

6. Navigate to the `config` subdirectory of the `$SIEBEL_ROOT` directory. For example, navigate to a location like `/Siebel/config`.
7. Start the Database Configuration Wizard, as described in *Starting the Siebel Database Configuration Wizard*. Enter a command similar to the following:

```
install_path/config/config -mode dbsrvr
```

In this path, install_path is the installation path for the installed Siebel Enterprise Server software.

The Database Configuration Wizard validates the existence of the`$SIEBEL_ROOT` directory.

8. Specify the following, and click Next:

   Siebel Server Directory. Specify the existing installation location of Siebel Server. This location is also referred to as the `$SIEBSRVR_ROOT` directory. For example, this directory might be `/export/home/Siebel/siebsrvr`.

   Siebel Database Server Directory. Specify the existing installation location of Siebel Database Configuration Utilities. This location is also referred to as the `DBSRVR_ROOT` directory. For example, this directory might be `/export/home/Siebel/dbsrvr`.
9. On the RDBMS Platform screen, select one of the following platforms, and click Next:

   IBM DB2 UDB for Linux UNIX Windows

   IBM DB2 UDB for z/OS

   Oracle Database Enterprise Edition
10. On the Siebel Database Operation screen, select Install Database from the list, and click Next.
11. On the Select Installation Operation screen, select Install Siebel Database to install a new Siebel database, or select Add a Language to an Existing Siebel Database, and then click Next.

    The Install Siebel Database option creates the Siebel schema and inserts seed data and Siebel Repository data for the primary language at the end of the Configuration Wizard session.

    If you instead selected Add a Language to an Existing Siebel Database, then go to Step 19. You must have included the language when you installed the Siebel Server and Database Configuration Utilities. For more information, see *Installing and Deploying Siebel CRM with Multiple Languages*.
12. On the next screen, confirm that you want to install a new Siebel database, or indicate that you already have a Siebel database.
13. On the Siebel User/Role Creation screen, confirm that you (or a database administrator) have already run the grantusr.sql script to set up table owner and administrator accounts.

    > **Note:** The table owner and administrator accounts must be in place or you will not be able to complete the installation of the tables, indexes, and seed data in the Siebel database. For more information, see *Creating Table Owner and Administrator Accounts*.

    - If you already ran grantusr.sql, then select the corresponding option, and click Next.
    - If you did not already run grantusr.sql, then exit the Database Configuration Wizard, and run the script now. When the script has finished executing, restart the Database Configuration Wizard.
14. (Oracle Database only) Specify the following, and click Next:

    Siebel Database Table Owner. Type the name of the Siebel database table owner or the account that will own the Siebel objects.

    Oracle SQLNet Connect String. Provide the name of the Oracle schema qualifier or table owner name and the SQLNet connect string for Oracle Database.
15. (IBM DB2 or IBM DB2 for z/OS only) Specify the following, and click Next:

    DB2 Instance Path. Provide the IBM DB2 client installation path, such as `/home/sblqa1`, where `sblqa1` contains the `sqllib` subdirectory.

16. (IBM DB2 only) Specify the following, and click Next:

    Siebel Database Table Owner. Type the name of the Siebel database table owner or the account that will own the Siebel objects.

    IBM DB2 Database Alias. Provide the name of the IBM DB2 UDB schema qualifier or table owner name and the database alias for IBM DB2.
17. (IBM DB2 for z/OS only) Specify the following, and click Next:

    Siebel Database Table Owner. Type the name of the Siebel database table owner or the account that will own the Siebel objects.

    IBM DB2 Database Alias. Provide the name of the IBM DB2 UDB schema qualifier or table owner name and the database alias for IBM DB2.

    Current IBM DB2 SQL ID / Group Name. Provide the name of the IBM DB2 UDB SQL ID or group name.
18. On the Database Encoding screen, identify the appropriate database encoding method, and click Next:

    UNICODE Database

    Non-UNICODE Database

    > **CAUTION:** Choose the correct option for your database to prevent installing the wrong data types. The database cannot create Unicode data types on a non-Unicode page setting, so check this setting carefully before choosing this option.

    > **Note:** Before you select languages in the next two steps, review the information about code pages in *Planning RDBMS Installation and Configuration*.
19. On the Base Language screen, specify which language serves as the primary language (base language) for the Siebel database.

    > **Note:** If you installed a single Siebel language on the Siebel Server where you also installed Database Configuration Utilities, then that language is assumed for a new database installation. Neither the Base Language nor the Language Selection screen appears. Go to Step 21.

    The languages listed in this screen are those for which you previously installed Siebel languages on the Siebel Server where you also installed Database Configuration Utilities. Do one of the following, and click Next:

    - If you are installing a new Siebel database, then specify the language that will be the primary language. Seed data and Siebel Repository data will be installed for this language. Go to Step 21. (The Language Selection screen does not appear in this case.)
    - If you are adding a language to an existing Siebel database, then specify the language that you already installed as the primary language. Go to the next step to specify the additional language to install, from the Language Selection screen.

20. (Add Language case only) On the Language Selection screen, select the language that you want to install in the Siebel database, for an Add Language operation.

    Note: For a new database install case, this screen does not appear. Go to the next step.

    The languages listed in this screen are those for which you previously installed Siebel languages on the Siebel Server where you also installed Database Configuration Utilities.

    Select the language that you want to add. Seed data will be installed for this language. You must import the Siebel Repository for this language separately (using the Add Language to an Existing Repository task), as described in *Importing a Siebel Repository Manually into the Siebel Database*.

    Some of the remaining screens do not apply for an add-language operation. Go to the next applicable step.
21. Specify the following, and click Next:

    Database User Name. Type the user name of the Siebel administrator; for example, `sadmin`.

    Database Password. Type the password for the Siebel administrator.

    Database Password (confirm). Retype the password to confirm it.
22. Specify the following, and click Next:

    Database Table Owner. Type the name of the Siebel database table owner or the account that will own the Siebel objects.

    Database Table Owner Password. Type the password for the Siebel database table owner.

    Database Table Owner Password (confirm). Retype the password to confirm it.
23. (Oracle Database only) On the Register External Oracle DB ODBC Driver screen, specify this option if you will use the Siebel Connector for Oracle Applications to exchange data with your back office system, and your back office system stores data in an Oracle Database.
24. (IBM DB2 only) Specify the following, and click Next:

    Index Table Space Name. Type the name for your index tablespace; for example, `SBL_INDX`. (Tablespace names must not contain spaces; underscores are allowed.)

    4K Table Space Name. Type the name for your 4-KB tablespace; for example, `TBS_4K`. (Tablespace names must not contain spaces; underscores are allowed.)

    16K Table Space Name. Type the name for your 16-KB tablespace; for example, `TBS_16K`. (Tablespace names must not contain spaces; underscores are allowed.)

    32K Table Space Name. Type the name for your 32-KB tablespace; for example, `TBS_32K`. (Tablespace names must not contain spaces; underscores are allowed.)
25. (Oracle Database only) Specify the following, and click Next:

    Index Table Space Name. Type the name for your index tablespace; for example, `INDEX01`. (Tablespace names must not contain spaces; underscores are allowed.)

    Table Space Name. Type the name for your tablespace; for example, `DATA01`. (Tablespace names must not contain spaces; underscores are allowed.)

26. (Oracle Database only) On the Siebel License Key Option screen, specify that you will provide the license key data later, and click Next.

    The license keys for Siebel CRM base applications must be activated using the License Key Activation utility, as described in *Activating License Keys*.
27. (Oracle Database only) On the Oracle Parallel Index screen, specify the statement describing your database environment, and click Next:
    - Does not use the Oracle Parallel Indexing option
    - Uses the Oracle Parallel Indexing option

    If you have a single-CPU environment, then choose the first option. If you have a multiple-CPU environment, then choose the first or second option, depending on whether you are running your Oracle Database with parallel indexing on.
28. Specify the following, and click Next:

    Security Group ID / Grantee. Type the name for your security group. Specify `SSE_ROLE`.
29. On the Log Output Directory screen, accept the default log directory or enter a new directory name, and click Next.

    By default, logging occurs in the `$SIEBSRVR_ROOT/log/operation/output` directory. In this path, operation corresponds to the operation that you are performing, such as *install* for an Install Database operation or *install_lang* for an Add Language operation.

    Click Help to view the following message, which asks if you are ready to apply configuration changes to the Siebel database now, using the Siebel Upgrade Wizard, or will do so later:

    ```
    Configuration is complete: your output will be saved under $SiebelRoot/siebsrvr/
    bin/master_<process>.ucf. Would you like to deploy the process you configured to
    the database now or later?
    ```

    > **Note:** Check the *Siebel CRM Update Guide and Release Notes* on My Oracle Support for the current release. If this document describes any tasks that you must perform to modify the generated SQL files, then modify the generated SQL files as necessary before you execute the SQL in the Siebel Upgrade Wizard.

30. Choose one of the following, and click Next:
    - **Yes apply configuration changes now.** Choose this option if you want the generated SQL files to be executed in the Siebel Upgrade Wizard.
    - **No I will apply configuration changes later.** Choose this option if you do *not* want the generated SQL files to be executed in the Siebel Upgrade Wizard. In this case, you must run the Upgrade Wizard manually at a later time in order to execute the generated SQL files.

    If you choose to apply configuration changes later, then the command line to apply the configuration later is:

    ```
    $SIEBEL_ROOT/siebsrvr/bin/srvrupgwiz /m master_operation.ucf
    ```

    For the purposes of this procedure, operation is either *install* or *install_lang*.
31. Review the values that you entered on the previous screens in the Database Configuration Wizard:
    - If you have to change any values, then click Back and Next to access the screens on which to change entries, and then to return to this screen.
    - Alternatively, you can click Cancel to cancel the Database Configuration Wizard session, then restart the wizard and enter different values.

- When you have verified your input, click Next. Doing so generates SQL files that you will then apply to the database. When all of the SQL files have been generated, a message appears, stating `The configuration changes were applied successfully.` Click OK.

If you chose No in Step 30, then the Siebel Upgrade Wizard does not appear. You must run it manually, after reviewing the generated SQL files.

If you chose Yes in Step 30, then the Siebel Upgrade Wizard appears.

> **Note:** When you run the Siebel Upgrade Wizard, if a program or system error occurs, then you can rerun the wizard, starting at the point at which the wizard failed. For more information, see *Siebel Database Upgrade Guide* .

When the Siebel Upgrade Wizard finishes running, the Siebel Upgrade Wizard Complete screen appears. You have now finished installing the Siebel database components.

32. After the Siebel Upgrade Wizard has finished running, click Exit to exit the Database Configuration Wizard.

Activating License Keys

License keys for Siebel CRM base applications that are provided in seed data in the Siebel database, or that were provided in prior releases, are inactive by default. License keys entered by customers for previous releases are unchanged. The License Key Activation utility is provided for activating or deactivating the license keys that you require. You run this utility to add any additional license keys after installing a new Siebel database, running Incremental Repository Merge (for migration installations), or completing a full database upgrade.

You can find license key information for Siebel CRM at Oracle's license codes site. For the Siebel license keys, see

http://licensecodes.oracle.com/siebel.html

The License Key Activation utility is supported on all operating systems and databases for Siebel CRM.

This task is a step in *Roadmap: Installing Siebel CRM for a New Deployment*.

To start the License Key Activation utility

1. On the computer where you installed Siebel Server, navigate to the following location:

   `SIEBSRVR_ROOT\bin`

2. (UNIX) Source either siebenv.sh or dbenv.sh in your environment.
   This step is necessary because licensekeymodule.sh accesses Siebel database via ODBC.
3. Run the following program, according to your operating system:
   - (Windows) licensekeymodule.bat
   - (UNIX) licensekeymodule.sh
4. Enter valid data for the following fields:
   - **Siebel Server Location.** The installation path for this Siebel Server.
   - **ODBC DSN.** The ODBC data source for the Siebel database.

- **Table Owner.** The table owner for the Siebel database.
- **Username.** The user name for logging into the Siebel database.
- **Password.** The password for this user.
- **DB Platform.** The RDBMS platform, either ORACLE, DB2UDB, DB2390, or (Windows) MSSQL.
- **Log folder.** The folder in which the log file licenseKeys.log is created. This log file shows database connection information for troubleshooting purposes, and lists all of the license keys that were activated or deactivated in each session.

5. Click Login.

   The license key activation screen appears, which lists Siebel CRM license keys.
6. For each license key module whose activation status you want to change, click the Active Flag check box to activate or deactivate this license key.
7. To apply your selections to the Siebel database, click Apply. Or, to reset any changes you have made in this screen, or since you last clicked Apply, click Reset.

   > **CAUTION:** After you have clicked Apply, the Reset button does not reset activation settings to their original state. However, you can change the activation status and click Apply again.

8. Click X to exit the utility.

Related Books

Siebel Database Upgrade Guide

Setting Up Your Environment to Support Global Time Zone

Global deployments typically span multiple time zones, making it difficult to manage time-sensitive information that must be exchanged among customers and employees working around the world. You can use the Global Time Zone feature to monitor the transfer of tasks between sites in different time zones.

The Global Time Zone feature converts and stores date and time data, using the Universal Time Coordinated (UTC) standard. UTC is equivalent to Greenwich Mean Time, but without daylight savings time.

> **CAUTION:** Oracle does not support new Siebel CRM deployments that do not use UTC. For existing deployments, if you do not want to deploy with UTC or are not ready to do so, then you must review the documentation for this feature, which is primarily located in *Siebel Global Deployment Guide* .

Make sure that the time is synchronized on all of your computers.

If you intend to operate your deployment with the Global Time Zone feature enabled, then you can optionally set the operating system of your database servers to UTC time, or its equivalent.

For restrictions on using the Global Time Zone feature on IBM DB2 for z/OS, see *Implementing Siebel Business Applications on DB2 for z/OS* .

Note: The Global Time Zone parameter (Universal Time Coordinated system preference) is enabled (set to TRUE) by default. If you do not want to enable the Global Time Zone feature, then you must reset this system preference to FALSE by navigating to the Administration - Application screen, and then the System Preferences view.

Related Books

Siebel Global Deployment Guide

Implementing Siebel Business Applications on DB2 for z/OS

Populating the Siebel File System

After you complete installing Database Configuration Utilities and installing the Siebel database, you must populate the Siebel File System. Specific files needed to use the Siebel File System, such as correspondence templates and Siebel Marketing files, are provided with the Database Configuration Utilities software. A subdirectory called `files` is created when you install the Database Configuration Utilities.

The Siebel administrator must populate the `att` directory in the File System with these files after installing the Database Configuration Utilities, and before running the Siebel Web Client.

For detailed information about creating the Siebel File System, including setting the necessary permissions, see *Creating the Siebel File System*.

To populate the Siebel File System directory

1. Copy the appropriate files from the `files` subdirectory of the Database Configuration Utilities software to the `att` subdirectory of the Siebel File System.
2. Verify that the files are in the correct location.

Importing a Siebel Repository Manually into the Siebel Database

Installing the Siebel database, as described in *Installing the Siebel Database*, automatically imports the Siebel Repository for the primary language (only). For each nonprimary language in a multilingual deployment, a separate step to manually import the Siebel Repository is needed.

The content in this topic is provided for customers with multilingual deployments or who have some reason to manually import a Siebel Repository.

You can import the Siebel Repository using the Database Configuration Wizard. This task populates the Siebel Repository tables in the Siebel database with new object definitions. You import the Siebel Repository separately for each nonprimary language in a multilingual deployment. This task does not apply to the Siebel Repository for the primary language.

Regardless of how many Siebel applications that you are using (for example, Siebel Call Center, Siebel Sales, Siebel Service, and Siebel Marketing), you load the Siebel Repository tables only once for each language.

Note: When you import data into the Siebel Repository tables, a commit is performed once for each table into which Siebel Repository data is imported. Alternatively, you can adjust the commit frequency by including the command-line option `/z size_of_array` when the repimexp.exe utility is invoked. The transaction will not be committed until all of the data in the array has been inserted. (However, if the table contains fewer records than the specified array size would allow, then the transaction is committed after all of the data has been inserted.)

Some steps in the procedure in this topic pertain to the task of adding a language to an existing Siebel Repository, also using the Database Configuration Wizard. By adding a new language to a Siebel Repository, you populate rows of localized user interface strings for Siebel Repository objects, which allows Siebel CRM to display the UI in the new language.

In order to add a new language to an existing Siebel Repository, you must have installed its Siebel language on the Siebel Server using the Siebel CRM installer. Also, you must have added the language to the Siebel database. For more information, see *Installing the Siebel Database* and *Installing and Deploying Siebel CRM with Multiple Languages*.

To import the Siebel Repository

1. Start the Database Configuration Wizard, as described in *Installing the Siebel Database*. Respond to the wizard prompts in the same way that you did for that procedure, up to the Siebel Database Operation screen.
2. On the Siebel Database Operation screen, select Import/Export Repository, and click Next.
3. On the Select Repository Operation screen, choose one of the following options:

   Import Repository. This option imports the Siebel Repository for the first time with a primary (base) language. Click Next and go to the next step. *This task does not apply if you are importing Siebel Repository data into an existing Siebel Repository for any nonprimary language.*

   Add Language to an Existing Repository. This option adds a new language to your existing Siebel Repository (imports Siebel Repository data for that language). Click Next and go to *Importing a Siebel Repository Manually into the Siebel Database*.

   Export Repository. This option exports the Siebel Repository data into a platform-independent file that can be sent to Oracle Global Customer Support for analysis, if needed. *This task is not described in this book.* For more information, see *Using Siebel Tools* and other documentation.
4. On the Import Selection screen, specify that you want to import the standard Siebel Repository, and click Next.
5. On the Language Selection screen, specify the language. For an import Siebel Repository operation, this is the primary language (base language), the first language installed in *Installing the Siebel Database*. For an Add Language operation, this is the language that you want to add to the Siebel Repository.

   The languages listed in this screen are those for which you previously installed Siebel languages on the Siebel Server.
6. On the ODBC Data Source Name screen, indicate the name for the ODBC data source, such as `sieb17_DSN`, and click Next.

   For more information about the ODBC data source, see *Planning RDBMS Installation and Configuration*.
7. On the Database User Name screen, indicate the following about your database, and click Next:

   Database User Name. Type the user name of the Siebel administrator, for example, `sadmin` for Oracle Database and IBM DB2, or `SADMIN` for Microsoft SQL Server.

   Database Password. Type the password for the Siebel administrator.

   Database Password (confirm). Retype the password to confirm it.
8. On the Database Table Owner screen, indicate the following about your database, and click Next:

Database Table Owner. The Siebel database table owner, or the account that will own the Siebel objects.

Database Table Owner Password. Type the Siebel database table owner password.

Database Table Owner Password (confirm). Retype the password to confirm it.

9. On the Import Repository Name screen, type the following values, and click Next:

   Import Repository Name. Accept the default name (Siebel Repository) or type another valid name.

   Repository File Name/Localized Repository File Name. If you are importing your Siebel Repository for the first time, then this field is named Repository File Name. If you are adding a language to an existing Siebel Repository, then this field is named Localized Repository File Name. Accept the default installation path and file name for this Siebel Repository, or type another valid installation path.

   For Oracle Database, go to the next step. For IBM DB2 or Microsoft SQL Server, go to Step 11.

10. (Oracle Database only) In the Oracle Parallel Index screen, specify the appropriate environment for your installation, and click Next:

    - I am not running with Parallel Indexing On
    - I am running with Parallel Indexing On

    If you have a single-CPU environment, then choose the first option. If you have a multiple-CPU environment, then choose the first or second option, depending on whether you are running your Oracle Database with parallel indexing on.

11. In the Log Output Directory screen, accept the default log directory, or enter a new directory name, and click OK.

    By default, logging occurs in the `SIEBSRVR_ROOT\log\operation\output` directory. In this path, operation corresponds to the operation that you are performing, such as imprep for Import Repository or imprep_lang for Add Language to an Existing Repository.

    The Configuration Parameter Review screen appears.

12. Review the configuration values that you entered on the previous Configuration Wizard screens:

    - If you want to change any values, then click Previous or Next to access the screens on which to change entries, and then to return to this screen.
    - When you have verified the configuration values, click Finish. A message box appears with the prompt:

      ```
      To apply the configuration now, press OK.
      To apply the configuration later, press Cancel.
      The command line to apply the configuration later is
      SIEBEL_ROOT\siebsrvr\bin\siebupg.exe /m master_operation.ucf
      ```

      For the purposes of this procedure, operation is either imprep or imprep_lang.

    The Siebel Upgrade Wizard appears, displaying the items to be executed or imported.

> **Note:** If a program or system error occurs and you have to rerun the Siebel Upgrade Wizard, then you can do so starting at the point at which the wizard failed. For more information, see *Siebel Database Upgrade Guide* .

13. To begin, click OK.

    A window appears, displaying information about Siebel Upgrade Wizard activities to import the Siebel Repository. The Upgrade Wizard displays a message when the operation is complete.

Related Topics

About Installing the Siebel Database

Verifying Installation for the Siebel Database

Troubleshooting Installation for the Siebel Database

Related Books

Using Siebel Tools

Siebel Database Upgrade Guide

6 Additional Postinstallation and Configuration Tasks

Additional Postinstallation and Configuration Tasks

This chapter describes some additional tasks that you might have to perform to complete configuring Siebel CRM. It includes the following topics:

- *Importing Siebns.dat into Zookeeper*
- *Starting the Siebel Gateway*
- *Stopping and Starting the Siebel Application Container*
- *Configuring Siebel Application Containers for Automatic Start on UNIX*
- *Starting the Siebel Server Service on Windows*
- *Configuring the Siebel Gateway for Automatic Start on UNIX*
- *Configuring the Siebel Server for Automatic Start on UNIX*
- *Managing Environment Variables on UNIX*
- *Preparing to Run Siebel Server Components*
- *Customizing the Application Container for Siebel Application Interface*
- *Installing and Deploying Siebel CRM with Multiple Languages*
- *Configuring Support for CalDAV or CardDAV Publishing*
- *Inactivating Workflow Processes*

Importing Siebns.dat into Zookeeper

Migrating content from Siebns utility can be executed standalone to move the content of siebns.dat to Zookeeper. The utility is integrated in the migration upgrade process that kicks off when a pre-Siebel CRM 17 env is upgrading to ISiebel CRM 17 and above Siebel deployment. The utility is present in the gtwysrvr/bin folder and can be executed at the command line. For standalone use a customer would first migrate the existing environment to Siebel CRM 17 and above version and then they can use on the same hosts by migrating the siebns.dat to ZK as is. It doesn't make any modification in any of the values. SMC will be used to deploy new servers with existing profiles (introspected as part of the migration upgrade process) and new parameters.

Using the arguments mentioned below.

Usage: `migratesiebns [ <args> ... ]`

Migrate siebns to zookeeper

```
-u <username> username (required)
-p [password] user's password (required) (to prompt for password, omit argument)
-g <host> Siebel registry host (required)
-t <port> Siebel registry port
-i <siebel server> siebns file path
```

```
-l <siebel server> log file path
```

Starting the Siebel Gateway

You must make sure that the Siebel Gateway system service is started when you do any of the following:

- Start any application container for Siebel CRM modules
- Start the Siebel Server system service
- Configure the Security profile
- Create and deploy the Siebel Enterprise profile
- Create and deploy a Siebel Server profile
- Create and deploy a Siebel Application Interface profile
- Create and deploy a Siebel Migration profile
- Operate any of the Siebel applications
- Remove the configuration for any of the Siebel CRM modules mentioned. (These are tasks that you might perform when you are uninstalling the software.)

Note: It is recommended that you review the Siebel Gateway log files, which are found in the `SIEBEL_GATEWAY_ROOT\LOG` directory.

To start the Siebel Gateway manually on Windows

1. Navigate to Control Panel, Administrative Tools, and then Services.
2. If the Siebel Gateway Name Server system service is not started, then click Action, and then Start.

   Note: For compatibility with prior releases, the actual name of the system service is Siebel Gateway Name Server (in contrast to the current usage of the name Siebel Gateway for the module itself).

To verify that the Siebel Gateway has started on UNIX

- Navigate to the `$SIEBEL_ROOT/gtwysrvr` directory and source the following environment setup file. Use one of the following commands, depending on the type of shell that you use:

  C shell

  ```
  source siebenv.csh
  ```

  Bourne or Korn shell

  ```
  ./siebenv.sh
  ```

  This step sets the Siebel environment variables and path information required to use Siebel utilities.

  To configure the Siebel environment shell script to source automatically whenever a Siebel administrator logs on, specify the shell as the default for administrator accounts. Then, add this command to the startup file for the administrator's account.

To verify that the Siebel Gateway started successfully and is running on UNIX

1. Enter the following command:

   ```
   list_ns
   ```

   If the Siebel Gateway Name Server system service is running, then you receive a response like the following, which shows the pid (process ID) of the running process:

   ```
   started at Mon May 01 20:06:55 2017, pid: 4024, autostart: yes
   ```

   If the Siebel Gateway Name Server system service is stopped, then you receive a response like this:

   ```
   stopped at Mon May 01 20:06:55 2017, autostart: no
   ```

2. If the Siebel Gateway Name Server system service is stopped, then do the following:
   - Execute the `start_ns` command to start the Siebel Gateway Name Server system service. For more information, see *Siebel System Administration Guide* .
   - Execute `list_ns` to verify that it is operating correctly.

Related Topics

Installing Siebel CRM Using the User Interface

Configuring Siebel Application Containers for Automatic Start on UNIX

Configuring the Siebel Gateway for Automatic Start on UNIX

Configuring the Siebel Server for Automatic Start on UNIX

Starting the Siebel Server Service on Windows

Verifying Installation for Siebel CRM

Verifying That the Siebel Gateway Has Started

Stopping and Starting the Siebel Application Container

If it is deemed necessary, then you can stop (shut down) and restart the Siebel application container using the methods in this topic. The application container for Siebel Gateway, Siebel Server (for Configuration Agent), Siebel Application Interface, and any other application container hosting Siebel CRM functionality are automatically started after installation. If application containers are not running, then you can perform the manual tasks listed in the troubleshooting sequence shown in this topic.

Application containers (Apache Tomcat) provided in installations of Siebel Gateway, Siebel Server (Configuration Agent), and Siebel Application Interface run as a service on Microsoft Windows, while the corresponding system process is named tomcat8.exe. On UNIX operating systems, the system process for Apache Tomcat is named javaw.

Note the following behavior for Microsoft Windows and for UNIX operating systems:

- On Microsoft Windows, when the computer restarts, for each applicable Siebel CRM installation, the Siebel Gateway system service (Siebel Gateway Name Server) starts automatically, followed by the application container for all installed Siebel CRM modules, followed by the Siebel Server system service, if also present.
- On UNIX operating systems, you can configure autostart for application containers, as described in *Configuring Siebel Application Containers for Automatic Start on UNIX*. You can also configure autostart for Siebel Gateway or Siebel Server system services, as described in *Configuring the Siebel Gateway for Automatic Start on UNIX* and *Configuring the Siebel Server for Automatic Start on UNIX*.

CAUTION: If you think you might need to shut down and then restart an application container, then, before you attempt to restart it, you must first check that the application container has shut down completely, for example, by using the Service Control Manager or the Task Manager in Windows. Otherwise, a port conflict failure might impede proper startup of the application container. Sometimes, examining logs might be the only way to indicate the nature of the problem.

Note: Decisions about whether to install Siebel CRM modules together or separately using the Siebel CRM installer must take into account the fact that modules installed together share the application container. Install modules separately to eliminate such dependencies and maintain availability for your Siebel CRM modules. If the Siebel Gateway and Siebel Server are collocated, then they share the same application container and do not require separate operations to stop or start the application container.

This topic includes the following information:

- *Troubleshooting Task Sequence When Siebel Application Containers Are Not Running*
- *Stopping the Siebel Application Container*
- *Starting the Siebel Application Container*

Related Topics

Starting the Siebel Gateway

Configuring Siebel Application Containers for Automatic Start on UNIX

Starting the Siebel Server Service on Windows

Customizing the Application Container for Siebel Application Interface

Verifying Installation for Siebel CRM

Troubleshooting Installation and Configuration for Siebel CRM

Related Books

Siebel System Administration Guide

Troubleshooting Task Sequence When Siebel Application Containers Are Not Running

When any of the application containers are not running, follow all applicable steps as follows. This topic is part of *Stopping and Starting the Siebel Application Container*.

1. Stop the application container for the Siebel Application Interface.
2. Stop the Siebel Server system service.
3. Stop the application container for the Siebel Server (Configuration Agent), if the Siebel Gateway and the Siebel Server are not collocated.
4. Stop the application container for the Siebel Gateway.
5. Stop the Siebel Gateway system service.
6. Start the Siebel Gateway system service.
7. Start the application container for the Siebel Gateway.
8. Start the application container for the Siebel Server (Configuration Agent), if the Siebel Gateway and the Siebel Server are not collocated.
9. Start the Siebel Server system service.
10. Start the application container for the Siebel Application Interface.

Stopping the Siebel Application Container

To shut down the application container, do one of the following. This task is part of *Stopping and Starting the Siebel Application Container*.

- **Microsoft Windows.** Run the Service Control Manager (services.msc), locate the service named like Apache Tomcat 9.0 SiebelApplicationContainer_internal_OracleHome_BuildNumber (for Siebel Enterprise components) or Apache Tomcat 9.0 SiebelApplicationContainer_external_OracleHome_BuildNumber (for Siebel Application Interface), and click the Stop button on the toolbar.

  In the preceding service name, OracleHome is the Oracle home for the Siebel CRM installation (such as SES or SIEBEL_AI) and BuildNumber is the software build number. The service name is automatically defined in this format at the time the service is created.

- **UNIX.** Navigate to the directory `$SIEBEL_HOME/applicationcontainer_internal/bin` (for Siebel Enterprise components) or `$SIEBEL_HOME/applicationcontainer_external/bin` (for Siebel Application Interface), and then execute the command `./shutdown.sh`.

  On UNIX operating systems, the shutdown.sh and startup.sh commands are provided in the following locations. (The Siebel Gateway and Siebel Server are assumed to be collocated and share the same application container, and so do not require separate operations to stop or start the application container.)

- **Siebel Enterprise.** `SIEBEL_ROOT/applicationcontainer_internal/bin`
- **Siebel Application Interface.** `SIEBEL_AI_ROOT/applicationcontainer_external/bin`

Starting the Siebel Application Container

To start the application container, do one of the following. This task is part of *Stopping and Starting the Siebel Application Container*.

- **Microsoft Windows.** Run the Service Control Manager (services.msc), locate the service named like Apache Tomcat 9.0 SiebelApplicationContainer_internal_OracleHome_BuildNumber (for Siebel Enterprise components) or Apache Tomcat 9.0 SiebelApplicationContainer_external_OracleHome_BuildNumber (for Siebel Application Interface), and click the Start button on the toolbar.

  In the preceding service name, OracleHome is the Oracle home for the Siebel CRM installation (such as SES or SIEBEL_AI) and BuildNumber is the software build number. The service name is automatically defined in this format at the time the service is created.

  > **Note:** Where you need to manually start any application container in your Siebel CRM deployment, do so only in the manner described here.

- **UNIX.** Navigate to the directory `$SIEBEL_HOME/applicationcontainer_internal/bin` (for Siebel Enterprise components) or `$SIEBEL_HOME/applicationcontainer_external/bin` (for Siebel Application Interface), and then execute the command `./startup.sh`.

  > **Note:** It is strongly recommended to configure all application containers in your Siebel CRM deployment to start automatically, or else configure none of them to start automatically. For more information, see *Configuring Siebel Application Containers for Automatic Start on UNIX*.

Configuring Siebel Application Containers for Automatic Start on UNIX

Complete the procedure that follows to configure the application containers to start automatically (autostart) on UNIX operating systems.

After making the changes described in this topic, the application container starts automatically when you restart your server and stops automatically when you shut down the server. For more information, see *Stopping and Starting the Siebel Application Container*.

> **Note:** It is strongly recommended to configure all application containers in your Siebel CRM deployment to start automatically, or else configure none of them to start automatically.

> **Note:** Before you perform this task, perform the tasks in *Configuring the Siebel Gateway for Automatic Start on UNIX*. Make sure the Siebel Gateway service starts before the application container starts. After you perform this task, perform the tasks in *Configuring the Siebel Server for Automatic Start on UNIX*.

Use the procedure in this topic to configure the application container to start automatically.

To configure the application container to start automatically on UNIX

1. Log on as root to a computer on which Siebel CRM was installed.
2. Copy the file SiebelApplicationContainer_OracleHome_BuildNumber.sh from `$SIEBEL_ROOT/applicationcontainer_external/bin` (for Siebel Application Interface) or `$SIEBEL_ROOT/applicationcontainer_internal/bin` (for other Siebel Enterprise components) to the standard library of autostart scripts for this computer, such as the `/etc/rc.d/init.d` directory or `/sbin/init.d` directory.

   OracleHome is the Oracle home for the Siebel CRM installation (such as SES or SIEBEL_AI) and BuildNumber is the software build number. `$SIEBEL_ROOT` is the full path to the root directory for the installed Siebel CRM software.
3. Using any text editor, edit the script SiebelApplicationContainer_OracleHome_BuildNumber.sh in the new location. Do one or more of the following:
   - Replace `$SIEBEL_ROOT` with the path to the Siebel CRM installation directory.
   - Modify hashplings in the script to reference the appropriate shell for this computer, such as C shell, Bash shell, and so on.
4. Set the appropriate permissions by executing commands like the following:

   ```
   chmod 755 /etc/rc.d/init.d/SiebelApplicationContainer_OracleHome_BuildNumber.sh
   chown root /etc/rc.d/init.d/SiebelApplicationContainer_OracleHome_BuildNumber.sh
   ```
5. Determine the default run level of the server by running the following command:

   ```
   who -r
   ```
6. Based on the run level determined, add symbolic links for the scripts.

   For example, if the default run level is 3, then create symbolic links under the `/etc/rc.d/rc3.d` directory, as in the following example:

   ```
   cd /etc/rc.d/init.d/rc3.d
   ln -s /etc/rc.d/init.d/SiebelApplicationContainer_OracleHome_BuildNumber
    S71SiebelApplicationContainer_OracleHome_BuildNumber
   ```

Related Topics

Stopping and Starting the Siebel Application Container

Configuring the Siebel Gateway for Automatic Start on UNIX

Configuring the Siebel Server for Automatic Start on UNIX

Verifying Installation for Siebel CRM

Troubleshooting Installation and Configuration for Siebel CRM

Related Books

Siebel System Administration Guide

Starting the Siebel Server Service on Windows

After you configure the Siebel Server and anytime you have restarted the server computer, the Siebel Server service starts automatically on Microsoft Windows. If you have stopped this service for any reason, then, in order to run the Siebel Server, you must restart it as described in this topic.

If you are clustering a particular server, then you must stop and start it using the Cluster Administrator, rather than by using the following procedure.

To manually start the Siebel Server service on Windows

1. Navigate to Control Panel, Administrative Tools, and then Services.
2. On the Services dialog box, scroll to the Siebel Server service. The name of the service has the following format:

   ```
   Siebel Server siebelenterprise_siebelserver
   ```

   where:

   - siebelenterprise is the name of your Siebel Enterprise
   - siebelserver is the name of the Siebel Server that you have just installed and configured
3. To start the service, select Action, then Start.

> **Note:** If the Siebel Server service does not start, then look in the log files for error messages. The installation log file is called SVRsetup.log and is located in the Siebel Server root directory. You can also check the file `log\siebel.SiebelServerName.log`.

Related Topics

Installing Siebel CRM Using the User Interface

Starting the Siebel Gateway

Verifying Installation for Siebel CRM

Configuring the Siebel Gateway for Automatic Start on UNIX

Complete the procedure that follows to configure the Siebel Gateway to start automatically (autostart) on UNIX operating systems. You run the Siebel Gateway service as the same nonroot user who installed the software.

After making the changes described in this topic for your operating system, the Siebel Gateway starts automatically when you restart your server and stops automatically when you shut down the server.

Note: After you perform this task, perform the tasks in *Configuring Siebel Application Containers for Automatic Start on UNIX*. Make sure the Siebel Gateway system service starts before the application container starts.

The Siebel Gateway must be the first service to start, and the last to shut down, among all of the servers in the Siebel Enterprise that are served by that Siebel Gateway.

This topic includes the following information:

- *Configuring the Siebel Gateway for Automatic Start on AIX*
- *Configuring the Siebel Gateway for Automatic Start on HP-UX*
- *Configuring the Siebel Gateway for Automatic Start on Linux or Oracle Solaris*

Related Topics

Installing Siebel CRM Using the User Interface

Starting the Siebel Gateway

Configuring Siebel Application Containers for Automatic Start on UNIX

Configuring the Siebel Server for Automatic Start on UNIX

Verifying Installation for Siebel CRM

Verifying That the Siebel Gateway Has Started

Setting Permissions and Ownership on UNIX

Configuring the Siebel Gateway for Automatic Start on AIX

Use the procedure in this topic to configure the Siebel Gateway to start automatically on AIX. This task is part of *Configuring the Siebel Gateway for Automatic Start on UNIX*.

To configure the Siebel Gateway to start automatically on AIX

1. Log on as root to the computer on which the Siebel Gateway was installed.
2. Execute the following command on a single line:

   ```
   mkitab "sieb_gtwyns:2:wait:su - SIEBEL_ACCOUNT -c \"$SIEBEL_HOME/bin/
   siebel_server start\""
   ```

   where:

   - SIEBEL_ACCOUNT is the nonroot user account installing Siebel CRM
   - `$SIEBEL_HOME` is the installation directory for the Siebel Gateway
3. Verify if the file `/etc/rc.shutdown` exists. If it does not exist, then create it and change the permissions:

   ```
   touch /etc/rc.shutdown
   chmod 755 /etc/rc.shutdown
   ```
4. Edit the file `/etc/rc.shutdown` and add the following command:

   ```
   /usr/bin/su - SIEBEL_ACCOUNT -c "$SIEBEL_HOME/bin/siebel_server stop"
   ```

where:

- SIEBEL_ACCOUNT is the nonroot user account installing Siebel CRM
- `$SIEBEL_HOME` is the installation directory for the Siebel Gateway

5. Save and close the file.

Configuring the Siebel Gateway for Automatic Start on HP-UX

Use the procedure in this topic to configure the Siebel Gateway to start automatically on HP-UX. This task is part of *Configuring the Siebel Gateway for Automatic Start on UNIX*.

To configure the Siebel Gateway to start automatically on HP-UX

1. Log on as root to the computer on which the Siebel Gateway was installed.
2. Copy the file siebel_server to the `/sbin/init.d` directory, as follows:

```
cp $SIEBEL_ROOT/bin/siebel_server /sbin/init.d
```

   In this command, `$SIEBEL_ROOT` is the full path to the Siebel root directory.
3. Using any text editor, edit `/sbin/init.d/siebel_server`. Do one or more of the following:
   - Replace $SIEBEL_GATEWAY_ROOT with the path to the Siebel Gateway installation directory.
   - Modify hashplings in the script to reference the appropriate shell for this computer, such as C shell, Bash shell, and so on.
4. Set the appropriate permissions by executing the following command:

```
chmod 755 /sbin/init.d/siebel_server
```

5. Create a soft link to `/sbin/rc3.d/S720siebel_server` from `/sbin/init.d/siebel_server` by executing the following command:

```
ln -s /sbin/init.d/siebel_server /sbin/rc3.d/S720siebel_server
```

6. Create a soft link to `/sbin/rc2.d/K320siebel_server` from `/sbin/init.d/siebel_server` by executing the following command:

```
ln -s /sbin/init.d/siebel_server /sbin/rc2.d/K320siebel_server
```

7. Create or edit `/sbin/init.d/siebel_server_nonroot` to add the following line:

```
/usr/bin/su - SIEBEL_ACCOUNT -c "/sbin/init.d/siebel_server $1"
```

   In this command, SIEBEL_ACCOUNT is the nonroot user account installing Siebel CRM.
8. Remove `/sbin/rc3.d/S720siebel_server` and `/sbin/rc2.d/K320siebel_server`, if they already exist.
9. Execute the following commands:

```
chmod 755 /sbin/init.d/siebel_server_nonroot
ln -s /sbin/init.d/siebel_server_nonroot /sbin/rc3.d/S720siebel_server
ln -s /sbin/init.d/siebel_server_nonroot /sbin/rc2.d/K320siebel_server
```

Configuring the Siebel Gateway for Automatic Start on Linux or Oracle Solaris

Use the procedure in this topic to configure the Siebel Gateway to start automatically on either Linux or Oracle Solaris. This task is part of *Configuring the Siebel Gateway for Automatic Start on UNIX*.

To configure the Siebel Gateway to start automatically on Linux or Oracle Solaris

1. Log on as root to the computer on which the Siebel Gateway was installed.
2. Copy the file siebel_server to the `/etc/init.d` directory, as follows:

   ```
   cp $SIEBEL_HOME/bin/siebel_server /etc/init.d
   ```

   In this command, `$SIEBEL_HOME` is the full path to the Siebel root directory.
3. Using any text editor, edit `/sbin/init.d/siebel_server`. Do one or more of the following:
   - Replace $SIEBEL_GATEWAY_ROOT with the path to the Siebel Gateway installation directory.
   - Modify hashplings in the script to reference the appropriate shell for this computer, such as C shell, Bash shell, and so on.
4. Set the appropriate permissions by executing the following command:

   ```
   chmod 755 /etc/init.d/siebel_server
   ```
5. Create a soft link to `/etc/rc3.d/S72siebel_server` from `/etc/init.d/siebel_server` by executing the following command.

   ```
   ln -s /etc/init.d/siebel_server /etc/rc3.d/S72siebel_server
   ```
6. Create a soft link to `/etc/rc2.d/K32siebel_server` from `/etc/init.d/siebel_server` by executing the following command:

   ```
   ln -s /etc/init.d/siebel_server /etc/rc2.d/K32siebel_server
   ```
7. Create or edit `/etc/init.d/siebel_server_nonroot` to add the following line:

   ```
   /usr/bin/su - SIEBEL_ACCOUNT -c "/etc/init.d/siebel_server $1"
   ```

   In this command, SIEBEL_ACCOUNT is the nonroot user account installing Siebel CRM.
8. Remove `/etc/rc3.d/S72siebel_server` and `/etc/rc2.d/K32siebel_server`, if they already exist.
9. Execute the following commands:

   ```
   chmod 755 /etc/init.d/siebel_server_nonroot
   ln -s /etc/init.d/siebel_server_nonroot /etc/rc3.d/S72siebel_server
   ln -s /etc/init.d/siebel_server_nonroot /etc/rc2.d/K32siebel_server
   ```

Configuring the Siebel Server for Automatic Start on UNIX

Complete the procedure that follows to configure the Siebel Server to start automatically (autostart) on UNIX operating systems. You run the Siebel Server service as the same nonroot user who installed the software.

> **Note:** Before you perform this task, perform the tasks in *Configuring the Siebel Gateway for Automatic Start on UNIX* and *Configuring Siebel Application Containers for Automatic Start on UNIX*. Make sure the Siebel Gateway service starts before the application container starts.

This topic includes the following information:

- *Configuring the Siebel Server for Automatic Start on AIX*
- *Configuring the Siebel Server to Start Automatically on HP-UX*
- *Configuring the Siebel Server to Start Automatically on Linux or Oracle Solaris*

Related Topics

Installing Siebel CRM Using the User Interface

Configuring the Siebel Gateway for Automatic Start on UNIX

Configuring Siebel Application Containers for Automatic Start on UNIX

Verifying Installation for Siebel CRM

Verifying That the Siebel Gateway Has Started

Troubleshooting Installation and Configuration for Siebel CRM

Configuring the Siebel Server for Automatic Start on AIX

Use the procedure in this topic to configure the Siebel Server to start automatically on AIX. This task is part of *Configuring the Siebel Server for Automatic Start on UNIX*.

To configure the Siebel Server to start automatically on AIX

1. Navigate to `$SIEBEL_ROOT/bin`.

   In this path, `$SIEBEL_ROOT` is the installation directory for the Siebel Server.

2. Using any text editor, open the file siebel_server. Edit the database environment variable as follows:

   Oracle Database

   a. Uncomment the line `#ORACLE_HOME=`.
   b. Enter the location for `$ORACLE_HOME`.
   c. Uncomment the line `#export ORACLE_HOME`.
   d. Add the following two lines:

   ```
   TNS_ADMIN=$tns_admin
   export TNS_ADMIN
   ```

   Ask your database administrator for the value of `$tns_admin` .

   IBM DB2

   - Add a line to execute `db2profile`, for example:

   ```
   . /home/db2v8aix/sqllib/db2profile
   ```

   Ask your database administrator for the location of `db2profile`.
3. Save and close the file.
4. Log on as root and execute the following command on a single line:

   ```
   mkitab "start_server:2:wait:su - SIEBEL_ACCOUNT -c \"$SIEBEL_ROOT/bin/
   siebel_server start\""
   ```

   where:

   - SIEBEL_ACCOUNT s the nonroot user account installing Siebel CRM
   - `$SIEBEL_ROOT` is the installation directory for the Siebel Server
5. Verify whether or not the file `/etc/rc.shutdown` exists. If it does not exist, then create it and change the permissions:

   ```
   touch /etc/rc.shutdown
   chmod 755 /etc/rc.shutdown
   ```

6. Edit the file `/etc/rc.shutdown`. Add the following command:

   ```
   /usr/bin/su - SIEBEL_ACCOUNT -c "$SIEBEL_ROOT/bin/siebel_server stop"
   ```

   where:

   - SIEBEL_ACCOUNT is the nonroot user account installing Siebel CRM
   - `$SIEBEL_ROOT` is the installation directory for the Siebel Server
7. Save and close the file.

Configuring the Siebel Server to Start Automatically on HP-UX

Use the procedure that follows to configure the Siebel Server to start automatically as a nonroot user on HP-UX. This task is part of *Configuring the Siebel Server for Automatic Start on UNIX*.

To configure the Siebel Server to start automatically on HP-UX

1. Log on as root.
2. Create or edit `/sbin/init.d/siebel_server_nonroot`. Add the following line:

   ```
   /usr/bin/su - SIEBEL_ACCOUNT -c "/sbin/init.d/siebel_server $1"
   ```

   In this command, SIEBEL_ACCOUNT is the nonroot user account installing Siebel CRM.
3. Remove `/sbin/rc3.d/S720siebel_server` and `/sbin/rc2.d/K320siebel_server`, if they already exist.
4. Execute the following commands:

   ```
   chmod 755 /sbin/init.d/siebel_server_nonroot
   ln -s /sbin/init.d/siebel_server_nonroot /sbin/rc3.d/S720siebel_server
   ln -s /sbin/init.d/siebel_server_nonroot /sbin/rc2.d/K320siebel_server
   ```

Configuring the Siebel Server to Start Automatically on Linux or Oracle Solaris

Use the procedure that follows to configure the Siebel Server to start automatically as a nonroot user on Linux or Oracle Solaris. This task is part of *Configuring the Siebel Server for Automatic Start on UNIX*.

To configure the Siebel Server to start automatically on Linux or Oracle Solaris

1. Log on as root.
2. Create or edit `/etc/init.d/siebel_server_nonroot`. Add the following line:

   ```
   /usr/bin/su - SIEBEL_ACCOUNT -c "/etc/init.d/siebel_server $1"
   ```

   In this command, SIEBEL_ACCOUNT is the nonroot user account installing Siebel CRM.
3. Remove `/etc/rc3.d/S72siebel_server` and `/etc/rc2.d/K32siebel_server`, if they already exist.
4. Execute the following commands:

   ```
   chmod 755 /etc/init.d/siebel_server_nonroot
   ln -s /etc/init.d/siebel_server_nonroot /etc/rc3.d/S72siebel_server
   ln -s /etc/init.d/siebel_server_nonroot /etc/rc2.d/K32siebel_server
   ```

Managing Environment Variables on UNIX

There are several environment variables that must be properly set on UNIX operating systems for optimal functioning of the Siebel Server, a few of which are shown in the following table. These include:

- Siebel environment variables
- UNIX operating system environment variables

Environment Variable	Purpose	Recommended Siebel Value
LANG	Language environment variable within the environment setup file. Sets the database server client environment appropriate to the language of installation. For more information about specifying locales, see *Specifying the Locale for Siebel CRM* and *Specifying the Locale for Your UNIX Operating System*.	Set to the language in which your database runs.
ODBCINI	Tells the ODBC driver manager which file to open to look for the ODBC data source and driver information. Set within the environment setup files, siebenv.csh (for C shell) and siebenv.sh (for Bourne or Korn shell), which are created during Siebel Enterprise Server and Siebel Server configuration. This variable always points to `$SIEBEL_HOME/sys/.odbc.ini`.	Do not modify.
RESOLV_MULTI	On Linux, for proper deployment of the Siebel Enterprise, this environment variable must be set on the computer where the Siebel Gateway application container runs. For example, you can define and export this variable in the .bash_profile file. If you do not set this variable as described, then deployment of the Enterprise profile fails.	Set to OFF.
SIEBEL_UNIXUNICODE_DB	Controls the ODBC driver manager responsible for connection to the RDBMS. Reset, if necessary, within each environment setup file before sourcing it.	Set to ORACLE or DB2. This value is set automatically.

Tuning UNIX Operating Systems for Siebel CRM

The Siebel environment variables $SIEBEL_HOME and $SIEBEL_ROOT must be set to enable execution of the Siebel Gateway and Siebel Server management utilities documented in *Siebel System Administration Guide* .

During Siebel Enterprise Server and Siebel Server configuration, the script files siebenv.csh (for C shell) and siebenv.sh (for Bourne or Korn shell) are automatically created in the `$SIEBEL_ROOT` directory. When sourced, these shell scripts set the environment variables.

If you have configured your Siebel Server to start manually, then you must source either siebenv.csh or siebenv.sh before starting the server. For this reason, you might want to add a call to the appropriate siebenv.* file to the logon files of all of the Siebel administrator UNIX accounts, so that these variables are set automatically whenever a Siebel administrator logs on.

If you want to change the default settings of these environment variables, then you must do so before you start the Siebel Server.

There are several UNIX system environment variables that you must set for proper functioning of the Siebel Server.

For detailed information about tuning the UNIX kernel and other configurable parameters for Siebel Server optimization, see *Siebel Performance Tuning Guide* . Many other environment variables are documented in *Siebel System Administration Guide* or other guides on *Siebel Bookshelf* .

Related Topics

Requirements for Installing and Configuring Siebel Enterprise Components

Specifying the Locale for Siebel CRM

Specifying the Locale for Your UNIX Operating System

Related Books

Siebel Performance Tuning Guide

Siebel System Administration Guide

Preparing to Run Siebel Server Components

This topic summarizes some of the tasks that you might have to do after installing and configuring Siebel Server and before you run Siebel Server components. It includes the following information:

- *Enabling and Disabling Server Component Groups*
- *Enabling and Disabling Language-Specific Application Object Managers and Adding Languages*
- *Synchronizing Batch Components*
- *Migrating Siebel Enterprise Server and Siebel Server Parameter Settings*
- *Performing Additional Tasks to Configure Siebel Servers*
- *Updating the CRC Checksum Setting*
- *About Installing Additional Siebel Servers for an Existing Siebel Enterprise*

You might have to perform some of these tasks on multiple Siebel Servers in order to configure them similarly for use in a load-balanced environment.

> **Note:** For detailed information about enabling and disabling server component groups, synchronizing batch components, creating server component job definitions, using Siebel Server Manager, and related tasks, see *Siebel System Administration Guide* .

Enabling and Disabling Server Component Groups

This task is part of *Preparing to Run Siebel Server Components*.

When you initially configured your Siebel Server, you specified which server component groups to enable. Before you deploy your Siebel Server, verify that the server component groups that you require are enabled. It is also recommended that you disable any server component groups that you do not require. Because you explicitly enabled server component groups when you configured the Siebel Server, you might or might not have to disable component groups after installation and configuration.

Some new server components were provided as of Siebel CRM 17.0. Also, some components have been removed or are no longer used. For example, additional customer applications for Siebel Open UI are provided that were previously desupported in Siebel CRM 16.0, but were available for standard interactivity in prior releases. As of Siebel CRM 16.0,

the Application Object Managers that in previous release were used for Siebel eSales, Siebel eCustomer, and Siebel eMarketing are now used for new customer applications built for Siebel Open UI.

Customers migrating from earlier releases also must consider which product changes are new to them because they were made in intervening releases. For example, in a prior release, some components moved from existing component groups into a new component group, Disconnected Mobile Synchronization (alias MobileSync). Consequently, if your deployment uses Siebel Mobile disconnected applications or uses Siebel Remote or Siebel Replication Manager, then you must enable the MobileSync component group, if you have not already done so, along with the HandheldSync or HandheldSyncSIS component group.

Product changes of these types introduce changed requirements for enabling server components and for deploying applications. The specific requirements that apply to your deployment depend on the products that you are using. Some requirements apply only for migration installations, and vary based on the Siebel CRM release that you are migrating from or the Siebel CRM version in effect when you first configured the software.

For example, for a new installation, you can enable any component groups using the Siebel Management Console. For a migration installation, if you do not reconfigure the Siebel Server and you have to enable any additional component groups, then you can do so manually after installation. In some cases, however, you might have to run certain scripts on the Siebel Server to make new products available.

Before performing a migration installation, also review parameter settings for server components affected by product changes, in case you need to reapply custom settings to components that moved, for example, or in case you must reconfigure the Siebel Server.

Several requirements apply to all migrations to the current release of Siebel CRM, as *Additional Tasks for Migration Installations*. For example, you must create the Siebel Application Interface profile and apply it to the installed instance of Siebel Application Interface.

For information about the requirements for particular Siebel CRM products, see product documentation on *Siebel Bookshelf* or elsewhere.

Related Topics

Additional Tasks for Migration Installations

Enabling and Disabling Language-Specific Application Object Managers and Adding Languages

Related Books

Siebel System Administration Guide

Siebel Mobile Guide: Disconnected

Siebel Remote and Replication Manager Administration Guide

Siebel Database Upgrade Guide

Enabling and Disabling Language-Specific Application Object Managers and Adding Languages

This task is part of *Preparing to Run Siebel Server Components*.

When you installed and configured the Siebel CRM software, you would have included one or more languages, as described in *Installing and Deploying Siebel CRM with Multiple Languages*. That topic also describes optional approaches to installing and deploying languages in a phased approach.

If you install the Siebel Server with multiple languages and specify these languages as deployed languages when you configure the Siebel Server, then language-specific Application Object Manager components are created for each deployed language. If you decide that you do not require Application Object Managers for one or more of these deployed languages, then you can use Siebel Server Manager to disable such components prior to deployment.

If you install multiple languages, then it is recommended that you deploy all of the installed languages when you configure the Siebel Server using the Siebel Management Console. As stated, you can disable any language-specific components that you do not require.

You must coordinate the language deployment tasks between your installations of Siebel Server and Siebel Application Interface.

For more information about performing tasks in the Siebel Management Console, see *Running the Siebel Management Console*.

Related Topics

Installing and Deploying Siebel CRM with Multiple Languages

Related Books

Siebel System Administration Guide

Siebel Global Deployment Guide

Synchronizing Batch Components

This task is part of *Preparing to Run Siebel Server Components*.

After installing and configuring the Siebel Server, you must synchronize any batch components before you can operate them. Do this after upgrading the Siebel database, where applicable. For more information about synchronizing batch components, see *Siebel System Administration Guide* .

Migrating Siebel Enterprise Server and Siebel Server Parameter Settings

This task is part of *Preparing to Run Siebel Server Components*.

Some Siebel Enterprise Server and Siebel Server parameter settings can be migrated from one Siebel application environment to another by using the configuration migration utility (cfgmerge). Other customized application data can be migrated by using Siebel Application Deployment Manager (Siebel ADM).

Related Books

Siebel System Administration Guide

Siebel Application Deployment Manager Guide

Performing Additional Tasks to Configure Siebel Servers

This task is part of *Preparing to Run Siebel Server Components*.

After installing and configuring a Siebel Server as described in this guide, you might have to perform some additional configuration tasks, such as setting parameters for the Siebel Enterprise, Siebel Server, or server components, such as Application Object Managers.

You perform these types of configuration tasks with Siebel Server Manager, whether by using views in the Administration - Server Configuration screen or by using commands at the srvrmgr command line. For more information about using Server Manager, see *Siebel System Administration Guide* .

Related Books

Siebel System Administration Guide

Updating the CRC Checksum Setting

This task is part of *Preparing to Run Siebel Server Components*.

If you are using a security adapter, then you might have to update the value of the CRC Checksum setting after installing Siebel CRM as a migration installation. The value must reflect the checksum value applicable to the security adapter library file, such as a DLL file on Microsoft Windows.

This task might be necessary if you previously determined to use checksum validation for your security adapter deployment, and set the value of the CRC Checksum setting. If a Siebel CRM release that you installed later as a migration installation included an updated security adapter library file, then checksum validation fails and Siebel Web Clients might not start. To prevent this result, you must update the value of the CRC Checksum setting.

For more information about specifying the CRC Checksum setting as part of configuring a security profile, see *Configuring a Security Profile*. See also *Siebel Security Guide* .

Related Books

Siebel Security Guide

About Installing Additional Siebel Servers for an Existing Siebel Enterprise

This task is part of *Preparing to Run Siebel Server Components*.

You can install and configure multiple Siebel Servers for a Siebel Enterprise.

Typically, before installing and configuring additional Siebel Servers for an existing Siebel Enterprise, you complete installation and configuration for the Siebel Application Interface.

Run the Siebel CRM installer to install each additional Siebel Server on another server computer, and run the Siebel Management Console to configure this Siebel Server.

You can also deploy a new Siebel Server based on an existing configured Siebel Server, using an existing Siebel Server profile you defined in Siebel Management Console.

After configuring a new Siebel Server, if this new Siebel Server adds a new component or language that is not available in any of the existing Siebel Servers, then you must perform corresponding Siebel Application Interface configuration tasks to make the applications on this Siebel Server available for Siebel native load balancing. In addition, you might have to modify the Siebel Application Interface profile for the installed instance.

Related Topics

Siebel CRM Download and Installation

Configuring Siebel CRM Server Modules

Customizing the Application Container for Siebel Application Interface

You can perform various configuration steps to customize the application container (Apache Tomcat) for Siebel Application Interface in your Siebel CRM deployment.

These tasks apply only to the application container for a deployed instance of Siebel Application Interface. They do not apply for the application container for any module that uses URLs for internal purposes only, such as Siebel Gateway or Siebel Server.

> **Note:** For each of the customizations described in this section, you must test your applications and verify that your customizations work as expected before you make these changes in your production systems.

This topic includes the following information:

- *Configuring the Common Logger for Siebel Application Interface*
- *Customizing URLs for Siebel CRM Applications*
- *Modifying the HTTPS Redirect Port or HTTP Port for Siebel Application Interface*
- *Compressing Static Content for Siebel Application Interface*
- *Load Balancing Siebel Application Interface by Configuring the workers.properties File*

Related Topics

Stopping and Starting the Siebel Application Container

Requirements for Installing and Configuring the Siebel Application Interface

Configuring the Common Logger for Siebel Application Interface

This task is part of *Customizing the Application Container for Siebel Application Interface*.

You can configure the log level for the Siebel Application Interface common logger, which is used for certain common logging tasks. This log cannot be configured using Siebel Management Console, unlike other types of logging for Siebel Application Interface described in *Configuring the Siebel Application Interface*.

If you plan to configure the log level for the common logger, it is recommended to do so after you finish configuring Siebel Application Interface using Siebel Management Console. After configuring the common logger, you must restart the application container, as described in *Stopping and Starting the Siebel Application Container*.

> **Note:** Static content (such as scripts and images) are separated into a directory named siebelwebroot. This siebelwebroot can support symbolic links to shared directory having custom files. This avoids adding one more entry into server.xml. Customers can ow create symbolic link to shared custom directories in siebelwebroot. To share custom files for multiple AI instances. Network speed and file locking issue must be considered when pursuing this option.

Related Topics

Stopping and Starting the Siebel Application Container

Related Books

Siebel System Monitoring and Diagnostics Guide

Customizing URLs for Siebel CRM Applications

This task is part of *Customizing the Application Container for Siebel Application Interface*.

You can perform various configuration steps to customize the URLs you use for Siebel CRM applications. The steps you perform for this task primarily involve editing files on the application container (Apache Tomcat) for Siebel Application Interface.

The keyword *siebel* is present in the Siebel CRM application URLs as a result of your deploying the siebel.war file on the application container, though it also exists in other configuration contexts. The particular customization you can make is to replace the keyword *siebel* in the Siebel URLs with a name suitable for your organization, such as *myorg*. Perform these steps after configuration and deployment tasks using Siebel Management Console. These steps do not involve renaming the siebel.war file.

You can specify a custom keyword when you install Siebel CRM in a new deployment. For more information, see *Installing Siebel CRM in a New Installation*.

The existing URL format is as follows:

```
https://host:port/siebel/app/application/lang
```

You can customize the Siebel Application Interface to use a different keyword, so the format would resemble the following:

```
https://host:port/myorg/app/application/lang
```

An example of a customized URL would be as follows:

```
https://server1:9000/myorg/app/callcenter/enu
```

To customize the URLs for Siebel CRM, use the steps in the procedure that follows.

To customize the URLs for Siebel CRM

1. Log in to the computer where you installed Siebel Application Interface.
2. Stop the application container, as described in *Stopping and Starting the Siebel Application Container*.
3. Navigate to the directory `SIEBEL_AI_ROOT\applicationcontainer_external\conf`.
4. Back up the server.xml file.
5. Open server.xml for editing, using a suitable text editing tool. For example, on Microsoft Windows you might use Notepad.
6. In server.xml, locate the <Host> tag, and modify this line to include the autoDeploy element, as follows:

   ```
   <Host name="localhost" appBase="webapps" unpackWARs="true" autoDeploy="true">
   ```

7. Under this <Host> tag, insert <Context> tags in lines like the following, substituting your preferred keyword for *myorg*. Include lines for all applicable languages.

   ```
   <Context docBase="${catalina.base}/siebelwebroot/images" path="/myorg/images" />
   <Context docBase="${catalina.base}/siebelwebroot/scripts" path="/myorg/scripts" />
   <Context docBase="${catalina.base}/siebelwebroot/smc" path="/myorg/smc" />
   <Context docBase="${catalina.base}/siebelwebroot/files" path="/myorg/files" />
   <Context docBase="${catalina.base}/siebelwebroot/htmltemplates" path="/myorg/htmltemplates" />
   <Context docBase="${catalina.base}/siebelwebroot/fonts" path="/myorg/fonts" />
   <Context docBase="${catalina.base}/siebelwebroot/migration" path="/myorg/migration" />
   <Context docBase="${catalina.base}/siebelwebroot/enu" path="/enu" />
   <Context docBase="${catalina.base}/siebelwebroot/enu" path="/myorg/enu" />
   <Context docBase="${catalina.base}/webapps/siebel" path="/myorg" />
   <Context docBase="${catalina.base}/webapps/ROOT/" path="/" />
   <Context path="/manager" docBase="${catalina.base}/webapps/manager" privileged="true" />
   ```

   If the `webapps` directory on the Application Interface also contains the subdirectories `host-manager`, `docs`, or `examples`, then also add <Context> lines like the following:

   ```
   <Context docBase="${catalina.base}/webapps/host-manager/" path="/host-manager" />
   <Context docBase="${catalina.base}/webapps/docs/" path="/docs" />
   <Context docBase="${catalina.base}/webapps/examples/" path="/examples" />
   ```

8. Save your updates to server.xml.
9. Restart the application container, as described in *Stopping and Starting the Siebel Application Container*.

   Next, you will update any referring links to the affected URLs.
10. Run Siebel Management Console. In the Siebel Application Interface profile, under the REST Inbound Defaults section, update the value for the REST Response Base URL to reflect the URL change. For example, change this URL:

    ```
    https://host:port/siebel/v1.0/
    ```

    as follows:

    ```
    https://host:port/myorg/v1.0/
    ```

    For more information, see *Configuring the Siebel Application Interface*.
11. Stop the application container, as described in *Stopping and Starting the Siebel Application Container*.
12. Modify the re-write.config file. Do the following:
    a. Navigate to the directory `SIEBEL_AI_ROOT\applicationcontainer_external\webapps\ROOT\WEB-INF`.

b. Back up the re-write.config file.
c. Open re-write.config for editing using a suitable text editor, such as Notepad in Windows.
d. Modify the following line, substituting *siebel* with the new keyword *myorg*, as you did in server.xml. For example, change this line:

```
RewriteRule ^/eai_enu/start.swe?(.*)$ /siebel/app/eai/enu/$1 [L]
```

as follows:

```
RewriteRule ^/eai_enu/start.swe?(.*)$ /myorg/app/eai/enu/$1 [L]
```

e. Make the same change in the lines for all other Siebel languages.
f. Save your updates to re-write.config.

13. Modify the ResourceSequence.txt file, which is used by the Siebel Migration application. Do the following:

a. Navigate to the directory `SIEBEL_AI_ROOT\applicationcontainer_external\webapps\siebel\WEB-INF`. Then back up the ResourceSequence.txt file, if it exists.
b. If ResourceSequence.txt does not exist, then navigate to `SIEBEL_AI_ROOT\applicationcontainer_external\webapps`, back up the siebel.war file, and extract ResourceSequence.txt into a directory of your choice (for temporary use).
c. Open ResourceSequence.txt for editing using a suitable text editor, such as Notepad in Windows.
d. Modify the following lines, substituting *siebel* with the new keyword *myorg*, as you did in server.xml. For example, change these lines:

```
AuthenticationURISuffix=/siebel/v1.0/service/Authentication%20Service%20For%20Migration/
AuthenticateUser
RepoUpgradeURISuffix=/siebel/v1.0/service/describe?searchspec=%5BName%5D%20LIKE%20%27Application
%20Migration%20Utility%20Service%27
```

as follows:

```
AuthenticationURISuffix=/myorg/v1.0/service/Authentication%20Service%20For%20Migration/
AuthenticateUser
RepoUpgradeURISuffix=/myorg/v1.0/service/describe?searchspec=%5BName%5D%20LIKE%20%27Application
%20Migration%20Utility%20Service%27
```

e. Save your updates to ResourceSequence.txt. Then replace this file in siebel.war.

When the application container is restarted, the updated version of ResourceSequence.txt is extracted into `SIEBEL_AI_ROOT\applicationcontainer_external\webapps\siebel\WEB-INF`.

14. Restart the application container, as described in *Stopping and Starting the Siebel Application Container*.
15. Perform the same steps for each instance of Siebel Application Interface in your deployment.

Modifying the HTTPS Redirect Port or HTTP Port for Siebel Application Interface

This task is part of *Customizing the Application Container for Siebel Application Interface*.

This topic describes the steps you can take to modify the HTTPS redirect port or HTTP port for a deployed instance of Siebel Application Interface.

To modify the HTTPS redirect port or HTTP port for a deployed instance of Siebel Application Interface

1. Using Siebel Management Console, delete the Application Interface deployment for which you are changing the HTTPS redirect port or HTTP port.

   For more information, see *Removing the Siebel Application Interface Configuration*.
2. Perform Steps 1 through 5 in *Customizing URLs for Siebel CRM Applications*.
3. In server.xml, replace all instances (three) of the HTTPS redirect port or HTTP port that you are changing with the new port number, which must be free.
4. Restart the application container, as described in *Stopping and Starting the Siebel Application Container*.
5. Restart Siebel Management Console.
6. In the Siebel Application Interface profile, under the REST Inbound Defaults section, update the value for the HTTPS redirect port or HTTP port.
7. Deploy the Application Interface profile with the new HTTPS redirect port or HTTP port.
8. Start a Siebel application that uses this port.

Compressing Static Content for Siebel Application Interface

This task is part of *Customizing the Application Container for Siebel Application Interface*.

This topic describes the steps you can take to specify HTML compression for the static content that resides on a deployed instance of Siebel Application Interface.

Enabling the setting HTTP 1.1-Compliant Firewall / Enable Web Compression for a Siebel Application Interface profile, which is described in *Configuration Settings for a Siebel Application Interface Profile*, compresses only dynamic content. If you plan to compress static Web content used with Siebel CRM applications, then you can configure the connector configuration as follows.

In the Siebel Application Interface installation, the static content are the Web artifacts for application configurations, which are located in `applicationcontainer_external\siebelwebroot`. The `siebelwebroot` directory contains subdirectories such as `files`, `fonts`, `htmltemplates`, `images`, `migration`, `scripts`, and `smc`.

> **CAUTION:** Setting properties to configure static compression affects all of the applications that are served on that server. You must make similar settings on the application container for all deployed instances of Siebel Application Interface. Before you enable static file compression, confirm your site requirements. Also carefully review the Apache Tomcat vendor portal for more information about configuring static file compression, and for any applicable issues or fixes.

To compress the static content for Siebel Application Interface

1. Stop the application container, as described in *Stopping and Starting the Siebel Application Container*.
2. Open the file server.xml, located in the directory `applicationcontainer_external/conf` on the Siebel Application Interface installation.
3. Find the connector port that is used to serve requests and add the following attributes. The following sample shows the connector port for HTTPS.

```
<Connector port=httpsport protocol="org.apache.coyote.http11.Http11NioProtocol"
 ...
 compressableMimeType="text/css,text/javascript,application/x-javascript,application/javascript"
```

```
useSendfile="off"
compression="on"
compressionMinSize="128"
connectionTimeout="20000"
noCompressionUserAgents="gozilla, traviata" />
```

4. Add the following properties, as needed: compressableMimeType, useSendfile, compression, compressionMinSize, connectionTimeout, and noCompressionUserAgents.
5. Save the server.xml file.
6. Restart the application container, as described in *Stopping and Starting the Siebel Application Container*.

Load Balancing Siebel Application Interface by Configuring the workers.properties File

This task is part of *Customizing the Application Container for Siebel Application Interface*.

This topic describes the initial configuration of the `workers.properties` file for load balancing Siebel Application Interface.

A load balancer acts as a reverse proxy and distributes network or application traffic across a list of available of servers. It works as a middleman between the client and the servers by accepting requests from clients and distributing the requests to the back-end servers based on the configured parameters. The `mod_jk` module of Apache HTTPd Server load balances Apache Tomcat servers and connects Apache Tomcat with Web servers using an AJP connector.

For more information about planning and managing load balancing for your deployment, see *Siebel Deployment Planning Guide* and *Siebel System Administration Guide* .

It's recommended that optimal usage of Siebel user session connections is maintained on Apache Tomcat.

To configure load balancing for Siebel Application Interface

1. Download and install the version of Apache HTTPd Server you want to install from the following location: http://tomcat.apache.org/
   For more information about installing Apache HTTPd Server, see Apache Tomcat documentation.
2. To configure the Listen port, go to the `HTTPDRootDir` directory and do the following:
   a. Using any text editor, open the httpd.conf file on the Web server.
   b. Locate the Listen section and add the HTTP port number:
      ```
      Listen port
      ```
   c. Save the `httpd.conf` file.
   d. Start Apache HTTPd Server.
   e. To check if Apache HTTPd Server is running, open a Web browser and enter the following URL: http://hostip:port
3. Download the Apache `mod_jk` module from the following location: `http://httpd.apache.org/download.cgi`
4. Save the `mod_jk.so` file and then place it in the `HTTPDRootDir` directory.
5. Create the following properties and log files:
   a. Create the `workers.properties` file and then place it in the `HTTPDRootDir` directory.

b. Create the `mod_jk.log` file and then place it in the `HTTPDRootDir` directory.

6. Modify the `httpd.conf` file, as follows:

   a. Add a Load statement to load the `mod_jk.so` file, as follows:

   ```
   # Load the mod_jk module
   LoadModule jk_module modules/mod_jk.so
   ```

   b. Add an IF statement to define the commands to be executed once the module is loaded, as follows:

   ```
   #Declare the module for use with the <IfModule directive> element.
   <IfModule jk_module>
    JkWorkersFile conf/workers.properties
    JkLogFile logs/mod_jk.log
    JkLogStampFormat "[%b %d %Y - %H:%M:%S] "
    JkRequestLogFormat "%w %V %T"
    JkLogLevel trace
    JkMount /* loadbalancer
    JkMount /Jkmanager status
   <IfModule>
   ```

   c. Save and close the `httpd.conf` file.

7. Define the list of Apache Tomcat workers that can accept requests by adding the following configuration to the workers.properties file:

   ```
   # Define workers
   worker.list=loadbalancer,status

   # Set properties for worker1
   worker.javacontainer1.type=ajp13
   worker.javacontainer1.host=<hostip>
   worker.javacontainer1.port=<ajportnumber>
   worker.javacontainer1.lbfactor=1
   worker.javacontainer 1.socket_keepalive=1
   worker.javacontainer1.socket_timeout=300

   # Set properties for worker2
   worker.javacontainer2.type=ajp13
   worker.javacontainer2.host=<hostip>
   worker.javacontainer2.port=<ajportnumber>
   worker.javacontainer2.lbfactor=1
   worker.javacontainer2.socket_keepalive=1
   worker.javacontainer2.socket_timeout=300

   # Set properties for loadbalancer
   worker.loadbalancer.type=lb
   worker.loadbalancer.balance_workers=javacontainer1,javacontainer2

   # Get statistics
   worker.status.type=status
   ```

   Note: For the `worker.workername` property, the `workername` must be same as the `JVMRoute` name defined in the Apache Tomcat server.xml file, and it must be unique across all Tomcat nodes being load balanced. Where tomcat1 and tomcat2 are load balanced and each has `javacontainer1,javacontainer2` as the value for `JVMRoute` in server.xml, the `workers.properties` file for both tomcat1 and tomcat2 must have parameters `worker.javacontainer1` and `worker.javacontainer2`.

8. Test the server by submitting an REST API request similar to the following example:

   ```
   http://hostip:port/siebel/v1.0/data/Account/Account/88-431RF
   ```

Related Books

Siebel REST API Guide

Installing and Deploying Siebel CRM with Multiple Languages

Multiple Siebel languages can be installed and deployed as part of Siebel CRM installation and configuration. Review this topic and related topics when planning multilingual deployments.

Language installation is part of the installation procedure for each installable Siebel CRM module. The task of adding and deploying new languages is described for Siebel Server, Siebel Application Interface, Siebel Web Client, and Siebel Tools.

Scenarios for deploying Siebel CRM with more than one language are described in *Scenarios for Installing and Deploying Multiple Languages*.

See also *General Requirements for Installing and Configuring Siebel Enterprise Components* and other relevant topics.

For a list of the languages supported by Siebel CRM, see the *Siebel Global Deployment Guide* , *Siebel System Administration Guide* , and other applicable documentation.

> **Note:** It's strongly recommended that you deploy, all of the languages that you expect to require, on each physical server.

You must run the Siebel Management Console and perform the tasks to redeploy languages to the Siebel Server. To do this, you create a new profile or clone the existing profile, delete the Siebel Server deployment item, and then deploy the Siebel Server with the new profile, specifying all of the languages to deploy. Finally, add the new languages to the Siebel Application Interface configuration. See also *Scenarios for Installing and Deploying Multiple Languages* and *Limitations for Installing Additional Languages*.

If you include multiple languages with a Siebel Enterprise Server installation, then you designate one language as the primary language to be used for server messages and logging. Typically, the same primary language is also used for the Siebel database, which cannot be changed after database installation.

If you install and deploy a new language in an existing installation, then you must add the language to the Siebel database and perform additional tasks using the Database Configuration Wizard, as described in *About Language Deployment Tasks Using the Database Configuration Wizard*.

> **Note:** Additional tasks in Siebel Tools might be required in order to fully deploy a language you have added. For more information, see the workspaces administration topics in *Using Siebel Tools* .

Related Topics

Scenarios for Installing and Deploying Multiple Languages

Limitations for Installing Additional Languages

About Language Deployment Tasks Using the Database Configuration Wizard

General Requirements for Installing and Configuring Siebel Enterprise Components

Related Books

Siebel Global Deployment Guide

Siebel System Administration Guide

About Language Deployment and Migration Installations

This topic is part of *Installing and Deploying Siebel CRM with Multiple Languages*.

All Siebel languages are included in the Siebel installation image on the network. To allow you to install any languages other than U.S. English (ENU), you also must include the media for non-ENU safeboot files in your installation image. You select languages in the Siebel CRM installer in order to install them.

When you perform a migration installation of Siebel CRM, files for all of the installed languages are automatically included. If you plan to perform a migration installation of Siebel CRM, and you also want to add languages to your existing deployment, then you've two options for how to do this, subject to limitations:

- **Add languages before installing Siebel CRM.** Subject to limitations for your existing installed Siebel CRM server and client software, it might be possible or desirable for you to install new languages before you install Siebel CRM. To determine if this option is possible for the currently installed software in your Siebel deployment and, if so, to determine what steps are required, see the documentation that applies to your currently installed Siebel software version for information about this version.
- **Add languages after installing Siebel CRM.** The installer for Siebel Enterprise Server and Siebel Application Interface supports adding languages to an existing installation of the same version, using GUI or silent mode installation methods. You'd perform this task after performing the migration installation tasks for these modules.

  For more information, see *Limitations for Installing Additional Languages* and *Installing and Deploying Additional Languages*.

For versions of the *Siebel Installation Guide* for your operating system for prior releases, see *Siebel Bookshelf* for each applicable release.

For example, if your current installation is Siebel CRM 8.1.1.10 on Microsoft Windows, then see *Siebel Installation Guide for Microsoft Windows* , version 8.1, Rev. D, and see *Siebel Maintenance Release Guide, Version 8.1.1.10*.

Related Topics

Limitations for Installing Additional Languages

Installing and Deploying Additional Languages

Importing Locale-Specific Data into the Siebel Repository

About Siebel CRM Releases

About Language Deployment on Siebel Server and Siebel Application Interface

This topic is part of *Installing and Deploying Siebel CRM with Multiple Languages*.

When you configure each installed instance of Siebel Server and Siebel Application Interface, you specify which languages to deploy, from among the installed languages. It is possible to deploy a subset of the installed languages. Which languages you deploy on a given server depends on the uses to which you will put that server.

Note the following points about language deployment on Siebel Server and Siebel Application Interface:

- On each Siebel Server, the set of deployed languages determines which language-specific components such as Application Object Managers are created on that server. It is recommended that you deploy all of the languages that are installed in the enterprise and optionally disable Application Object Managers for particular languages where they are not required.
- On each instance of Siebel Application Interface, you must deploy the superset of all of the languages that are required by all of the Application Object Managers that connect to it. It is recommended that you deploy all of the languages that are installed in the enterprise.

Related Topics

Configuring Siebel CRM Server Modules

Scenarios for Installing and Deploying Multiple Languages

About Language Deployment Tasks Using the Database Configuration Wizard

This topic is part of *Installing and Deploying Siebel CRM with Multiple Languages*.

In addition to software installation and configuration tasks, for installed languages other than the primary language for the Siebel database, you must perform the following operations, using the Database Configuration Wizard. You perform these tasks after installing the Siebel database.

- **Adding the language (including seed data) to the Siebel database.** Perform this task once for each nonprimary language that you install.
- **Importing the Siebel Repository into the Siebel database.** Perform this task once for each nonprimary language that you install.
- **Running the MLOV (multilingual LOV) conversion utility.** Perform this task once after installing multiple languages initially and again after adding one or more languages.

Related Topics

Installing the Siebel Database on the RDBMS

Installing the Siebel Database

Importing a Siebel Repository Manually into the Siebel Database

About Language Deployment for Unshipped Languages

This topic is part of *Installing and Deploying Siebel CRM with Multiple Languages*.

If you are localizing Siebel CRM to use a language for which a Siebel language is not provided, then see also the topics about localizing an unshipped language in *Siebel Global Deployment Guide* . In particular, note that installing Siebel CRM to update an existing installation requires that you copy certain files to keep your unshipped language deployment up to date for the current release.

Limitations for Installing Additional Languages

This topic is part of *Installing and Deploying Siebel CRM with Multiple Languages*.

Although it is strongly recommended to install all of the languages that you might require (whether or not you initially deploy them all), you can also add languages to an existing installation. However, several limitations apply.

> **Note:** If you plan to perform a migration installation of Siebel CRM, and you also want to add languages to your existing deployment, then you might be able to install the languages before you install the software. For more information, see *About Language Deployment and Siebel installation and Migration Installations*.

Limitations for installing additional languages are as follows:

- You can install additional languages for existing installations by using the installers of the same version only. For example, you can install the software as a new installation or as a migration installation, then later use the same installers to add languages at the new version level.
- You can add languages by using either GUI or silent mode installation operations.
- You cannot generate response files for an operation to add languages. You must use an edited copy of one of the response files that are provided for Siebel Enterprise Server, Siebel Application Interface, Siebel Web Client, or Siebel Tools. For the location of these response files, see *Installing and Deploying Additional Languages*.
- You cannot add languages as part of performing a migration installation. However, you can add languages after the migration installation.

Related Topics

About Language Deployment and Siebel installation and Migration Installations

Installing and Deploying Additional Languages

Uninstalling Siebel CRM

Scenarios for Installing and Deploying Multiple Languages

This topic is part of *Installing and Deploying Siebel CRM with Multiple Languages*.

Consider the following scenarios for installing and deploying multiple Siebel languages:

- **Install and deploy all of the languages that you require once.** For customers who are installing and deploying languages provided with the Siebel CRM release and *do not* plan to deploy other languages at a later date. This option is easiest but might not be suitable for phased language deployments.
- **Install all of the languages that you will require, but deploy some of these languages later.** For customers who are installing and deploying languages provided with the Siebel CRM release and plan to deploy some of the installed languages at a later date. This option is recommended for phased language deployments.
- **Install and deploy additional languages in an existing deployment.** For customers who are installing and deploying languages provided with the Siebel CRM release and plan to install and deploy additional languages at a later date. This option might be suitable for some phased language deployments. You can install languages at any time to meet changing business needs. However, deploying languages is easier if you include them when you first install and configure the Siebel CRM software.

Related Topics

Installing and Deploying All Required Languages Once

Installing All Required Languages but Deploying Some Languages Later

Installing and Deploying Additional Languages

Installing and Deploying All Required Languages Once

This task is part of *Installing and Deploying Siebel CRM with Multiple Languages*.

This scenario is intended for customers who are deploying languages that were provided with Siebel CRM and who are installing all of the languages that their enterprise will require during initial installation and configuration. Include all of the required languages with your initial installation and deploy them when you configure the installed Siebel CRM software.

> **Note:** You perform the tasks in this topic in conjunction with topics such as *Installing Siebel CRM Using the User Interface*. Perform all applicable steps that are documented in more detail in these and other topics. These tasks are also part of the roadmap topics in *Overview of Installing Siebel CRM*.

To deploy multiple languages shipped with Siebel CRM

1. Run the Siebel CRM installer and install Siebel Enterprise Server components on all of the applicable servers, with all of the languages that you will require for the enterprise.
2. Install Siebel Application Interface on all of the applicable servers, with all of the languages that you will require for the enterprise.
3. Install the Siebel database. This task installs seed data for the primary language into the Siebel database.
4. Using the Siebel Management Console, create and deploy the Security profile, the Siebel Gateway profile, and the Siebel Enterprise profile.
5. Create and deploy the Siebel Server profile. During Siebel Server configuration, specify to deploy all of the installed languages.
6. Create and deploy the Siebel Application Interface profile. During Siebel Application Interface configuration, specify to deploy all of the installed languages.
7. For each additional installed (nonprimary) language that you are deploying for the first time: add the language to the Siebel database and import the Siebel Repository. Also run the MLOV conversion utility (once).

Installing All Required Languages but Deploying Some Languages Later

This task is part of *Installing and Deploying Siebel CRM with Multiple Languages*.

This scenario is intended for customers who are deploying languages that were provided with Siebel CRM but who plan to deploy some of these languages at a later date on particular instances of Siebel Server or Siebel Application Interface. Include all of the required languages with your initial installation.

There are two variations of this scenario that can achieve the desired result. Review each option and decide which one works best for your situation. See also *Installing and Deploying All Required Languages Once* for details for some tasks.

> **Note:** You perform the tasks in this topic in conjunction with topics such as *Installing Siebel CRM Using the User Interface*. Perform all applicable steps that are documented in more detail in these and other topics. These tasks are also part of the roadmap topics in *Overview of Installing Siebel CRM*.

Deploying Multiple Languages Now and Later (Option 1)

This task is part of *Installing and Deploying Siebel CRM with Multiple Languages*.

Use the following procedure to deploy multiple languages now and at a later date. This procedure represents Option 1. Compare it with Option 2.

To deploy multiple languages now and at a later date (Option 1: preferred)

1. Perform tasks similar to those described in *Installing and Deploying All Required Languages Once*.
    - Install Siebel CRM on all of the modules in your Siebel CRM deployment, including all of the languages that you will deploy now and at a later date.
    - When you configure each Siebel Server and Siebel Application Interface, specify that you want to deploy all of the installed languages.
2. For each Siebel Server, disable any language-specific Application Object Manager components that you do not yet require.
3. When you are ready to deploy additional installed languages, re-enable any language-specific Application Object Managers that you previously disabled.
4. For each additional installed (nonprimary) language that you are deploying for the first time: add the language to the Siebel database and import the Siebel Repository. Also run the MLOV conversion utility (once).

Deploying Multiple Languages Now and Later (Option 2)

This task is part of *Installing and Deploying Siebel CRM with Multiple Languages*.

Use the following procedure to deploy multiple languages now and at a later date. This procedure represents Option 2. Compare it with Option 1.

To deploy multiple languages now and at a later date (Option 2)

1. Perform tasks similar to those described in *Installing and Deploying All Required Languages Once*. In this case, it is assumed that you did not deploy all of the installed languages.
    - Install Siebel CRM on all of the modules in your Siebel CRM deployment, including all of the languages that you will deploy now and at a later date.
    - When you configure each Siebel Server or Siebel Application Interface, optionally specify to deploy only a subset of the installed languages.
2. When you are ready to deploy additional installed languages, perform the following for each Siebel Server:
    a. Run the Siebel Management Console, as described in *Starting the Siebel Management Console*.
    b. Create a new Siebel Server profile or clone the existing profile.
    c. Delete the Siebel Server deployment item.
    d. Deploy the Siebel Server with the new profile, specifying all of the languages to deploy.
3. Perform the following for each Siebel Application Interface:
    a. Run the Siebel Management Console, as described in *Starting the Siebel Management Console*.
    b. Modify the existing Siebel Application Interface profile to add support for required component-language combinations, as described in *Creating a Siebel Application Interface Profile*.
    c. Specify the additional languages to deploy.
    d. Save the profile.
4. For each additional installed (nonprimary) language that you are deploying for the first time: add the language to the Siebel database and import the Siebel Repository. Also run the MLOV conversion utility (once).

Installing and Deploying Additional Languages

This task is part of *Installing and Deploying Siebel CRM with Multiple Languages*.

This scenario is intended for customers who did not install all of the Siebel languages and who later determine that they require additional supported Siebel languages that were not previously installed and deployed.

This topic describes adding new languages to your existing Siebel CRM deployment and existing physical resource allocation. As part of expanding language support, you might also install Siebel CRM modules on new servers that include new languages.

For an existing installation of Siebel CRM modules, you can run the same installer in either GUI or silent mode to add any available and supported languages that you require that were not previously installed.

For detailed requirements and other information about running Siebel installers and Siebel Management Console, see *Requirements for Installing and Configuring Siebel CRM* and other applicable parts of this guide.

Perform the tasks in this topic in conjunction with the installation instructions for each installable module, including those for using silent mode. Perform all applicable steps, including those that are documented in more detail in other topics.

For more information about performing silent installations, see *Installing Siebel CRM Using Silent Mode*.

Note: Where you are adding languages to an existing installation, the primary language for the Siebel database will already have been determined when you installed the Siebel Enterprise Server software and the Siebel database. Any language that is to serve as the primary language for a new Siebel database must be included when you first install the Siebel Enterprise Server software.

Response File Parameters

The response files that you use if you install additional languages in silent mode include parameters that specify information about your existing installation, the network image, the new languages you are installing, and so on. Review the parameters in each response file to determine the parameters for which you must provide values. All values must be enclosed in double-quotes.

For any operation to install new languages, you must do the following:

- Specify values for the ORACLE_HOME and ORACLE_HOME_NAME parameters, which correspond to values that you specified when you performed the original installation. These settings identify the installation to which you are adding languages.
- Specify a value for the s_shiphomeLocation parameter, representing the location of the `Disk1` directory in the network image for the Siebel CRM module that you are installing.
- Specify a value for the FROM_LOCATION parameter, representing the location of the products.xml file within the `Disk1\stage` directory in the network image for the Siebel CRM module that you are installing.
- Specify a value for the selectedLangs parameter, representing the new language or languages that you are installing.

For Siebel Enterprise Server only, you must set the following parameters to true for each installed module:

- b_isGatewayInstalled (if Siebel Gateway is installed)
- b_isSiebsrvrInstalled (if Siebel Server is installed)
- b_isDBInstalled (if Siebel Database Configuration Utilities is installed)
- b_isEAIInstalled (if EAI Connector is installed)

To install and deploy additional languages on existing installations

1. Perform tasks similar to those described in *Installing and Deploying All Required Languages Once* or *Installing All Required Languages but Deploying Some Languages Later*. In this case, it is assumed that you did not install and deploy all of the languages.
2. Review *Limitations for Installing Additional Languages*.
3. For each applicable Siebel Server, when you are ready to deploy the newly installed languages, and any other languages that were not previously deployed, do the following:
    a. Run the Siebel Management Console, as described in *Starting the Siebel Management Console*.
    b. Create a new Siebel Server profile or clone the existing profile.
    c. Delete the Siebel Server deployment item.
    d. Deploy the Siebel Server with the new profile, specifying all of the languages to deploy.
4. For each applicable Siebel Application Interface, do the following:
    a. Run the Siebel Management Console, as described in *Starting the Siebel Management Console*.
    b. Modify the existing Siebel Application Interface profile to add support for required component-language combinations, as described in *Creating a Siebel Application Interface Profile*.

c. Specify the additional languages to deploy.
d. Save the profile.

5. For each additional installed (nonprimary) language that you are deploying for the first time: add the language to the Siebel database and import the Siebel Repository. Also run the MLOV conversion utility (once).
6. Validate the installation after it has completed.
7. Repeat the previous steps, where applicable, to install and deploy the new languages for each applicable installed Siebel CRM module.

> **Note:** Add the languages for all of the applicable server installations before you add the languages for Siebel Web Client and Siebel Tools, where needed.

Related Topics

Installing Siebel CRM Using the User Interface

Installing Siebel CRM Using Silent Mode

Limitations for Installing Additional Languages

Installing and Deploying All Required Languages Once

Installing All Required Languages but Deploying Some Languages Later

Verifying and Troubleshooting Your Installation

Requirements for Installing and Configuring Siebel CRM

About Siebel CRM Releases

Importing Locale-Specific Data into the Siebel Repository

This task is part of *Installing and Deploying Siebel CRM with Multiple Languages*.

Sometimes it is necessary to update the text strings held in the repository for a particular language or for multiple languages. These text strings are held in the Symbolic String Model (SSM) table, and appear on-screen as labels for tabs, fields, and so on, across all of the Siebel applications.

Updates might be needed either because a string has been incorrectly translated for a particular language or because installing an update release has caused a new string or message to appear for all of the languages for which Oracle ships languages for Siebel CRM.

> **Note:** These instructions are provided for use with any releases to which they might apply. They might not apply for any particular Siebel CRM release or language.

The following procedure uses the Locale Management Utility (LMU), which is part of Siebel Tools, to import an LMU file. An LMU file might contain strings in just one language or in multiple languages. When running the Locale Management Utility, you can choose to import only the strings for the languages that you are using in your installation. In the procedure, when you are prompted for the file to import, specify the name of the LMU file provided for the applicable language.

> **Note:** Install Siebel CRM before you run the Locale Management Utility.

For more information about using the Locale Management Utility, see *Using Siebel Tools* and *Siebel Global Deployment Guide* .

To import strings and other locale-specific attributes into the repository

1. In Siebel Tools, choose Tools, then Utilities, and then Locale Management.
   The Locale Management Utility appears.
2. Select a source language and a target language.
   The source language is the language of the locale-specific data in the LMU file that you are importing. The target language is the Siebel language into which the strings will be imported.
3. Click the Import tab.
4. Enter the name of the LMU file from which you want to import locale-specific attributes.
   You can also click Browse to find and select the file.
5. Specify whether you want to mark records in the repository with the Redo flag that have changed since the export occurred.
   When the import occurs, the LMU compares the source language records in the repository with the source language records in the import file. If the records in the repository have changed since the export occurred, then they are marked with the Redo flag. This flag helps you identify records that might have to be retranslated.
6. Click Import.
7. After completing the import process, update the repository and deliver the updates, then restart the Application Object Manager component on the Siebel Server to see the results.

Related Books

Using Siebel Tools

Siebel Global Deployment Guide

Configuring Support for CalDAV or CardDAV Publishing

This topic provides information about configuring your Siebel CRM environment to support CalDAV or CardDAV publishing. CalDAV (Calendaring Extensions to WebDAV) is an internet standard designed to allow users to access and share calendar data on a remote server. CardDAV (vCard Extensions to WebDAV) is a client-server protocol designed to allow users to access and share address book or contact data on a server. These standards allow users to view external contact or calendar information in Siebel applications.

To enable CalDAV or CardDAV for your Siebel CRM deployment, the following requirements apply:

- On the computer or operating system instance where you installed Siebel Application Interface, Transport Layer Security (TLS) must be configured. For more information, see *Siebel Security Guide* .
- On the Siebel Server, the EAI component group must be enabled. For more information, see *Configuring the Siebel Server*. See also *Siebel System Administration Guide* and additional documentation on *Siebel Bookshelf* for Siebel Enterprise Application Integration.
- In the Siebel CRM applications, valid email addresses must be associated with all Employees and Contacts. For each mobile user, the email address defined in Siebel CRM must match the user's email address on the mobile device. For more information, see *Siebel Mobile Guide: Connected* or *Siebel Mobile Guide: Disconnected* . See also *Siebel Applications Administration Guide* .

- Users of mobile devices must set up CalDAV or CardDAV publishing on their devices in order to be able to access calendar or contact data. For more information, see *Siebel Mobile Guide: Connected* or *Siebel Mobile Guide: Disconnected* .
- Additional steps must be performed to enable dynamic mapping for CardDAV. Siebel CRM supports dynamic mapping for CardDAV on mobile devices running supported versions of iOS. CardDAV dynamic mapping defines the mapping between Siebel objects and the CardDAV properties. The Siebel CardDAV Sync module synchronizes the defined entities (address book and contact data) accordingly between a user's mobile device and Siebel CRM applications.

Related Books

Siebel Security Guide

Siebel System Administration Guide

Siebel Mobile Guide: Connected

Siebel Mobile Guide: Disconnected

Siebel Applications Administration Guide

Inactivating Workflow Processes

You can run the InactivateWFs.exe utility in development environments (Design Time Repository or DR) to inactivate workflow processes that are no longer required. The utility inactivates workflow processes that have not already been deployed or modified and which are not present in S_WFA_DPLOY_DEF (the old workflow deployment table). The InactivateWFs.exe utility inactivates workflow processess by setting the Inactive flag for the workflow processes to True and removes the corresponding compiled runtime object definition records. You might want to do this task after installing the current Siebel CRM release (and after running PostInstallDBSetup and related utilities).

The following procedure describes how to inactivate workflow processes that are no longer required.

To inactivate workflow processes

1. Run the InactivateWFs.exe utility from the command line using values appropriate to your database. This utility is located in the `siebsrvr\bin` or `Tools\bin` directory. The syntax to use is as follows:

```
InactivateWFs.exe -s <InstallationPath> -t <SiebelTableOwner> -u <TableOwnerUsername> -p <Password> -o
 <ODBCDataSource>
```

```
-d <RDBMSPlatform> -r <RepositoryName> -k <LogDirectory>
```

For example, for Oracle Database:

```
InactivateWFs.exe -s "../ses/siebsrvr" -t SIEBEL -u SIEBEL -p ********** -o SIEBEL -d ORACLE -r "Siebel
 Repository" -k "../ses/siebsrvr/log"
```

For example, for Microsoft SQL Server/Azure SQL Database Server:

```
 InactivateWFs.exe -s "..\ses\siebsrvr" -t dbo -u SIEBEL -p ********** -o Siebel_DSN -d MSSQL -r "Siebel
 Repository" -k "..\ses\siebsrvr\log"
```

For example, for DB2390:

```
InactivateWFs.exe -s "../ses/siebsrvr" -t SIEBEL -u SIEBEL -p ********** -o SIEBEL -d DB2390 -r "Siebel
 Repository" -k "../ses/siebsrvr/log"
```

2. Output appears that is similar to the following:

```
Stage 1 of 3 : Setting the Repository Id
Stage 2 of 3 : Find the WF data
Stage 3 of 3 : Delete the run time events
Inactivated Workflows : ../ses/siebsrvr/log/InactiveWFs_20200609_003106\InactivateWFs.out
```

If there are other errors, then review the log file(s) indicated in the error messages, resolve any issues, and then rerun the process.

Related Books

Siebel Business Process Framework: Workflow Guide

7 Installing Siebel Web Clients and Siebel Tools

Installing Siebel Web Clients and Siebel Tools

This chapter provides information about installing Siebel Web Clients and Siebel Tools and about starting Siebel applications. Additional tasks for the client computer are also described. This chapter includes the following topics:

- *About Installing Siebel Web Client or Siebel Tools*
- *Installing and Using Oracle Database SE2 for the Local Database*
- *About Workspaces in Siebel Tools and the Siebel Runtime Repository*
- *Installing Desktop Integration Siebel Agent on Client Computers*
- *Siebel Client Deployment Requirements When Using Firewalls or Proxy Servers*
- *Configuring Siebel CRM for Pop-Up Blocker Compatibility*
- *Logging in to Your Siebel Application*
- *Siebel Web Client Shortcuts and Startup Options*

About Installing Siebel Web Client or Siebel Tools

This chapter provides information about installing Siebel Web Client (for Siebel Mobile Web Client or Developer Web Client) and Siebel Tools. Before installing, review all documented requirements.

> **Note:** You use the same Siebel CRM installer to install Siebel Web Client or Siebel Tools that you use to install Siebel Enterprise Components. For details, see *Installing Siebel CRM Using the User Interface*, and use the procedure for new installation, update installation, or migration installation.

Note the following about Siebel Web Client and Siebel Tools:

- Each Siebel Web Client is designed to be used within a single Siebel Enterprise by a single local user.
- Each Siebel Tools client is designed to be used within a single Siebel Enterprise by a single local developer user.

  > **Note:** Siebel Web Tools is now the preferred alternative to the traditional Siebel Tools client. For more information, see *Using Siebel Tools* .

You install Oracle Database SE2 and Oracle Database Client with the Siebel Web Client for the local database on Siebel Mobile Web Client. See also *Installing and Using Oracle Database SE2 for the Local Database*.

The software that is installed on the user's computer for Siebel Mobile Web Client and Developer Web Client is identical. The Siebel CRM installer installs the Siebel Web Client software and supports the configuration of the Siebel Web Client for use as a Mobile Web Client. Configuration options are also provided for the Developer Web Client.

For general information about these client types, see *Siebel Deployment Planning Guide* . Information in this guide about installing the *Siebel Web Client* refers to either or both of these two client types. See also *Deploying Siebel Open UI* .

This chapter also includes information about installing additional modules on client computers, for use with any Siebel Web Client.

In general, installing Siebel Web Client or Siebel Tools does the following:

1. Checks the client computer to verify whether required components have already been installed.
2. Creates all of the required ODBC data sources.
3. Installs the Siebel CRM software and languages.
4. (Siebel Mobile Web Client) Installs Oracle Database SE2. For more information, see *Installing and Using Oracle Database SE2 for the Local Database*.
5. Creates Siebel shortcuts.
6. Writes data to installation log files.

Note: The Siebel Developer Web Client is also available, for administration, development, and troubleshooting usage scenarios only. The software that is installed on the user's computer for Siebel Mobile Web Client and Developer Web Client is identical. Configuration options are also provided for the Developer Web Client. You might choose to install the Developer Web Client with Siebel Tools. The Developer Web Client has specific requirements for connectivity to the enterprise database. Other than the Oracle Database Client, RDBMS products are not provided with Siebel CRM. You must obtain them based on your database-connection requirements.

Related Topics

Installation-Related Changes in Siebel CRM

Installation Tasks for Siebel CRM

Installing Siebel CRM Using the User Interface

Installing Siebel CRM Using Silent Mode

Additional Tasks for Migration Installations

Installing and Using Oracle Database SE2 for the Local Database

Verifying Installation for the Siebel Web Client

Verifying Installation for Oracle Database SE2

Uninstalling Siebel CRM

About Installing Siebel CRM

Requirements for Installing Siebel Web Clients

Requirements for Installing Siebel Tools

Related Books

Siebel Remote and Replication Manager Administration Guide

Using Siebel Tools

Configuring Siebel Business Applications

Configuring Siebel Open UI

Siebel Developer's Reference

Installing and Using Oracle Database SE2 for the Local Database

Oracle Database SE2 is provided as the database platform for the local database for Siebel Mobile Web Client. The installer for Siebel CRM installs Oracle Database SE2 software for Siebel Web Client (Siebel Mobile Web Client only). Oracle Database XE is no longer supported, and the sample database is no longer provided.

After installation, the local database on each remote client is initialized through the Siebel Remote server. This local database is also called the remote database.

This topic includes the following information:

- *Requirements for Installing and Using Oracle Database SE2*
- *About the Data Source for the Local Database*
- *About Migration Installations*
- *About the Local Database for Siebel Mobile Web Client*
- *About the Sample Database for Prior Versions of Siebel Mobile Web Client*
- *About Uninstalling Oracle Database XE*

Requirements for Installing and Using Oracle Database SE2

This topic is part of *Installing and Using Oracle Database SE2 for the Local Database*.

Platform, database, and other certifications for Siebel are found on My Oracle Support. For help searching for certifications, see *How to Check the Certification Matrix in the New My Oracle Support Site* (KB871388).

See also *Requirements for Installing Siebel Web Clients*.

- **Generate and extract a new local database.** After installing Siebel CRM server modules, you must run the Generate New Database (GenNewDb) and Database Extract (DbXtract) server components to generate and extract a new local database for your remote users. This database uses Oracle Database Standard Edition 2 (SE2). For more information, see *Siebel Remote and Replication Manager Administration Guide* .
- **Download Oracle Database SE2.** You must install Oracle Database SE2 with Siebel Web Client for the local database. To install Oracle Database SE2 on client computers, you download the associated .zip file from Oracle and include it in the Siebel CRM installation image before you install Siebel Web Client, if wasn't installed in a previous update.

  Obtaining this .zip file and installing Oracle Database SE2 is a one-time operation. When you install subsequent Siebel CRM releases, and Oracle Database SE2 at the same version is already installed, then it won't be reinstalled. Updated versions of Oracle Database SE2 could become available in the future and be required for Siebel Web Client. It's not supported to use an existing installation of Oracle Database SE2 outside of the Siebel installation.
- **Administrator privileges required.** Installing Oracle Database SE2 with Siebel Web Client requires administrator privileges on the computer on which you're installing.

- **Memory requirement.** Oracle Database SE2 requires at least 2 GB of RAM on the computer on which you're installing.
- **Postinstallation configuration required.** After installing Oracle Database SE2 for Siebel Web Client, you must perform the following configuration task:

  a. Navigate to `SIEBEL_CLIENT_ROOT\oui\bin`.
  b. Open the file UpdateCFG.bat, using an editor like Notepad.
  c. Modify the file to indicate the configuration file for the particular application you're running, such as uagent.cfg for Siebel Call Center. For example, you might update UpdateCFG.bat as follows to specify the uagent.cfg file:

     ```
     c:\Siebel\Client\jre\bin\java.exe -jar c:\Siebel\Client\oui\jlib\UpdateCFG.jar c:\Siebel\Client
     \BIN\enu\uagent.cfg false
     ```

  d. Run UpdateCFG.bat.
  e. Repeat steps 3 and 4, as needed, for all applications your users will run in this Siebel Mobile Web Client.
- **Configure automated deployment.** You can optionally configure automated deployment for the Siebel Web Client with Oracle Database SE2, using third-party deployment tools. Your automated deployment, for example, would:
  - Sign in to the client computer as a user with administrator privileges.
  - Install the Siebel Web Client along with Oracle Database SE2 and Oracle Database Client, using responses needed for this installation. See also *Installing Siebel CRM Using Silent Mode*.
  - Update the configuration files as a postinstallation step (run UpdateCFG.bat).
  - Do cleanup such as deleting old files, as needed.
- **Installation logging.** For Oracle Database SE2 installation logging details, see the log files for the Siebel Web Client installation. For more information, see *Verifying Installation for the Siebel Web Client*.
- **Verify installation.** After installing Oracle Database SE2 with the Mobile Web Client, verify your installation, as described in *Verifying Installation for the Siebel Web Client* and *Verifying Installation for Oracle Database SE2*.

About the Data Source for the Local Database

This topic is part of *Installing and Using Oracle Database SE2 for the Local Database*.

When you install Siebel Mobile Web Client for the current release of Siebel CRM, the LOCAL_SE data source is created, which you use for connections to a local database. The corresponding section of the Siebel application configuration file is [LOCAL_SE].

After the local database has been extracted for the user on the Siebel Remote server, then you initialize this database by logging in using the LOCAL_SE data source, as described in *Siebel Remote and Replication Manager Administration Guide* .

About Migration Installations

This topic is part of *Installing and Using Oracle Database SE2 for the Local Database*.

Before any migration installation tasks are performed, local database users must synchronize any changes that must be preserved. Oracle Database SE2 must then be installed with Siebel Web Client for the current release of Siebel CRM, and a new local database must be initialized after it has been extracted by Siebel Remote administrators, as with a new installation. A new data source is also created.

About the Local Database for Siebel Mobile Web Client

This topic is part of *Installing and Using Oracle Database SE2 for the Local Database*.

For information about initializing the local database for Siebel Mobile Web Client through Siebel Remote, see *Siebel Remote and Replication Manager Administration Guide* . The data in each local database is a subset of the server database, as determined by each user's visibility rules.

The user connects to the LOCAL_SE data source to initialize the local database. When logging in to a local database for the first time, the user is automatically prompted to connect to the Siebel Server and download the local database. Users must attempt this only after a local database has been extracted for them by the system administrator.

About the Sample Database for Prior Versions of Siebel Mobile Web Client

This topic is part of *Installing and Using Oracle Database SE2 for the Local Database*.

Oracle Database SE2 is used for the local database. Oracle Database XE and the sample database are no longer provided or supported for use with the Siebel Mobile Web Client.

To continue to use the sample database, you must install a Siebel Mobile Web Client for Siebel CRM 20.7 or earlier and install the sample database with Oracle Database XE. Do not update this client to any later Siebel CRM release. For details, see the applicable *Siebel Installation Guide* for Siebel CRM 20.7 or earlier. To install the sample database for Siebel Mobile Web Client in a new or migration installation, see the topic "Installing the Siebel Web Client" in the old *Siebel Installation Guide* and refer to any other relevant information about the sample database or Oracle Database XE. Connect to the SAMPLE_XE data source to log in to this sample database.

Where the sample database has been installed on the local computer at Siebel CRM 20.7 or earlier, you can log in and connect to the sample database, using predefined demonstration (demo) users. For example, CCHENG is a demo user for Siebel Call Center. Many other demo users are available in the sample database that illustrate a particular Siebel application as it would appear for a user with certain predefined responsibilities. For more information about these users, see *Siebel Demo Users Reference* on the *Siebel Bookshelf* for Siebel CRM 20.7 or earlier.

Shortcuts for all demo applications for demo users use the sample database. You can also create demo application shortcuts that specify the sample database and user and password information. For this example, you might define a shortcut like Siebel Call Center Demo - ENU as shown in the following:

```
"C:\Siebel\Client\BIN\siebel.exe" /c CFG_FILE_PATH\uagent.cfg /d sample /u CCHENG /p CCHENG
```

Note that the demo application shortcuts specify predefined demo users using `/u` and `/p` options.

About Uninstalling Oracle Database XE

This topic is part of *Installing and Using Oracle Database SE2 for the Local Database*.

As of Siebel CRM 20.8, when you install Siebel Web Client in an update installation, Oracle Database XE is shut down automatically and can no longer be used with Siebel CRM. After installation, you can uninstall Oracle Database XE. For information about uninstalling Oracle Database XE, see *Uninstalling Oracle Database XE*. See also *About Uninstalling Earlier Versions of Siebel CRM*.

About Workspaces in Siebel Tools and the Siebel Runtime Repository

Workspaces in Siebel Tools and Siebel Web Tools provide a developer user with a sandbox in which to make and test changes to repository data. This feature allows multiple developers to work against the same repository objects in the Siebel database. The feature makes sure that a developer's work is isolated from that of other developers making changes to either the same objects or other objects of the application. Repository updates made in the developer's environment can be published and tested without affecting other developers, until all updates are coordinated and published and delivered to the runtime repository for all Siebel Servers and Siebel Mobile Web Clients in test and production environments. For more information, see *Using Siebel Tools* .

As of Siebel CRM 17.0, the Siebel Repository File (SRF) is no longer applicable, and is replaced by the Siebel runtime repository in the Siebel database. As of Siebel CRM 20.3, the seamless repository framework supports nonextensible object files and a hybrid repository type.

> **Note:** Siebel Tools and Siebel Web Tools always use the standard Siebel Repository provided for the current release. You can optionally configure Siebel Application Object Managers and Siebel Mobile Web Client to use the same standard Siebel Repository for troubleshooting purposes. To do so, set the EnableSafeboot parameter to True for Siebel Application Object Managers and Siebel Mobile Web Clients. For Siebel Mobile Web Clients, you set this parameter in the [InfraObjMgr] section of the application configuration file, such as uagent.cfg for Siebel Call Center.

Related Topics

About Installing Siebel Web Client or Siebel Tools

About Database Updates for Siebel CRM

Related Books

Configuring Siebel Business Applications

Using Siebel Tools

Installing Desktop Integration Siebel Agent on Client Computers

This topic provides information about installing software on user computers that enables users to use certain features in Siebel applications that formerly were based on Java applets. This software is called Desktop Integration Siebel Agent, which is sometimes called DISA.

Customers can install and deploy Desktop Integration Siebel Agent on user computers for use with employee applications such as Siebel Call Center. Desktop Integration Siebel Agent supports inline editing of attachments, CTI hoteling, Send Email (F9) integration with Microsoft Outlook, and batch fulfillment printing.

For more information about installing and deploying Desktop Integration Siebel Agent, see *Desktop Integration Siebel Agent Guide* . For information about hoteling, see *Siebel CTI Administration Guide* . For more information about using Desktop Integration Siebel Agent for the Send Email (F9) integration with Microsoft Outlook.

> **Note:** Future Siebel CRM releases might include an updated version of Desktop Integration Siebel Agent, which might support additional features. If a new version is distributed in a Siebel CRM release, then the new version will be described in *Desktop Integration Siebel Agent Guide* .

Related Books

Desktop Integration Siebel Agent Guide

Siebel CTI Administration Guide

Siebel CRM Update Guide and Release Notes on My Oracle Support

Siebel Client Deployment Requirements When Using Firewalls or Proxy Servers

This topic describes Siebel client deployment requirements that apply when you use firewalls or proxy servers, such as reverse proxy servers. It applies to all Siebel applications that are deployed inside the firewall or using a proxy server.

This topic includes the following information:

- *Deploying Siebel Applications Accessed Through a Firewall*
- *Bypassing the Proxy Server for Local Addresses*

Deploying Siebel Applications Accessed Through a Firewall

This task is part of *Siebel Client Deployment Requirements When Using Firewalls or Proxy Servers*.

When deploying Siebel applications that are accessed through a firewall, if you are planning to enable compression for traffic between the Siebel Application Interface and the Web browser, then your firewall must support the HTTP 1.1 protocol.

If your firewall does not support HTTP 1.1, then do not set HTTP 1.1-Compliant Firewall / Enable Web Compression to False for the Siebel Application Interface. In addition, disable HTTP 1.1 for proxy connections on each client computer. Follow instructions for your browser for this purpose. For example, for Microsoft Internet Explorer, you uncheck the option Use HTTP 1.1 Through Proxy Connections.

> **Note:** For more information about deploying Siebel CRM with firewalls, and about compression settings for Siebel Application Interface, see *Configuring the Siebel Application Interface*. See also *Siebel Security Guide* and *Siebel System Administration Guide* .

Bypassing the Proxy Server for Local Addresses

This task is part of *Siebel Client Deployment Requirements When Using Firewalls or Proxy Servers*.

If your LAN uses a proxy server, then adjust your browser settings to bypass the proxy server for local (internal) addresses. This setting provides better performance for Siebel employee applications deployed on the LAN, and is required for the Siebel Mobile Web Client.

Follow instructions for your browser for this purpose. For example, for Microsoft Internet Explorer, you check the options Use a Proxy Server for Your LAN and Bypass Proxy Server for Local Addresses. For Siebel Web Clients, you also specify the proxy server address and port number.

> **Note:** For customer applications, this setting does not apply. The proxy server setting applies only on a LAN.

Configuring Siebel CRM for Pop-Up Blocker Compatibility

Pop-up blocking software can sometimes cause Siebel applications not to work, because such software might block required Siebel application functionality. Such blocking software typically blocks processes or pop-up windows that originate from particular computer host names or domain names.

Siebel applications that you run on a LAN (Siebel Web Client) or that run on local client computers (Siebel Mobile Web Client) might be affected by pop-up blockers. Configuring the Siebel application to use a fully qualified domain name (FQDN), as described in this topic, might prevent such problems and allow you to use pop-up blockers safely. See also *Configuring the Siebel Application Interface*. It is strongly recommended to use the fully qualified domain name feature.

For example, the URL http://ebiz.example.com uses FQDN, while http://ebiz does not.

> **Note:** If you have configured FQDN for your Siebel applications, and if conflicts with pop-up blockers persist, then it might be necessary to uninstall the pop-up blocking software or any applications in which such functionality is built in.

Parameters to configure fully qualified domain names are located in multiple places. How you set these parameters depends on the Siebel client that you are using. Specified host names or domain names must meet all of the applicable requirements, for example, underscores cannot be used.

- **Siebel Web Client.** To use FQDN for this client type, set Configure Fully Qualified Domain Name to True when you configure the Siebel Application Interface instance, and set Fully Qualified Domain Name to the fully qualified domain name. For more information, see *Configuring the Siebel Application Interface*.

  For example, you might set the fully qualified domain name to corp.example.com or server1.corp.example.com.

- **Siebel Mobile Web Client.** To use FQDN for this client type, configure the following parameters in the Siebel application configuration file, such as uagent.cfg for Siebel Call Center, on each local client computer.

  ```
  [Siebel]
  EnableFQDN = True
  ```

  EnableFQDN is set to True by default for the Siebel Mobile Web Client.

  If you do not also specify the FQDN parameter, then the URL is constructed automatically. For example, the FQDN for CCHENG might be constructed as ccheng.corp.example.com. Optionally, you can explicitly provide similar information, by using the FQDN parameter, as follows:

  ```
  FQDN = hostname.primaryDNS.domainsuffix
  ```

  where:

  - hostname is the name of the local client computer
  - primaryDNS is the primary part of the domain name (such as example)
  - domainsuffix is the domain type (such as com)

  For example, you might set FQDN to ccheng.corp.example.com.

  > **CAUTION:** When you explicitly configure an FQDN for use with the Siebel Mobile Web Client, you must specify the local computer name. The localhost string from the default Siebel Mobile Web Client URL (which is used when EnableFQDN is False) cannot be used as an element in an FQDN. The localhost string is only functional when used by itself, with no additional qualifying elements.

Logging in to Your Siebel Application

This topic provides basic instructions for logging in to your Siebel application using one of the Siebel Web Client types. It includes the following information:

- *Before Logging in to Your Siebel Application*
- *Logging in Using the Siebel Mobile Web Client*
- *Logging in Using the Siebel Web Client*

Before Logging in to Your Siebel Application

This task is part of *Logging in to Your Siebel Application*.

This topic contains information that you must know before you log in to your Siebel application. Review the following issues:

- User rights to read and write in Siebel Web Client installation directories are required for running the Siebel Mobile Web Client or Developer Web Client. For information about setting user rights, consult the operating system manuals for the version of Microsoft Windows on which the application is installed.
- After the Siebel database has been installed, as described in *Installing the Siebel Database on the RDBMS* and the server modules have been configured and you activate the license keys, users can access the Siebel CRM applications. For information about activating the license keys in the Siebel database, see *Activating License Keys*.
- The user name and password used to log in to Siebel applications must be those of an employee with a valid position and division defined in the Siebel database.

  You must log in to Siebel applications (using the Server database option in the login screen) as the Siebel administrator before anyone else can log in. Log in using the SADMIN user name or using other credentials as defined by your database administrator. Then you or other administrators can set up more users as employees with defined positions and responsibilities and other settings suitable for your access control policies. For more information about setting up employees, see *Siebel Security Guide* .
- For the Siebel Developer Web Client, you must edit the application configuration file, such as uagent.cfg for Siebel Call Center, as follows:
  - In the [Siebel] section of the file, add the parameter ServerDbODBCDataSource, and set it to the ODBC data source that you are using for the server database.
  - In the [ServerDataSrc] section of the file, set the parameter ConnectString to the necessary value. For Oracle Database, set it to the alias defined in the tnsnames.ora file.
- Tabbed browsing is supported with Siebel applications for one active session. Multiple Web pages can be opened using tabbed browsing, but only one tab can have a live connection to a Siebel application at a given time.

Logging in Using the Siebel Mobile Web Client

This task is part of *Logging in to Your Siebel Application*.

You can use the Siebel Mobile Web Client to log in to the Siebel application.

To log in to your Siebel application using the Siebel Mobile Web Client

1. Double-click one of the Siebel shortcuts in the program folder, such as Siebel Call Center for the language you are using.
2. Log in using a valid user ID and password.

3. Specify the Local database, and click OK.

   Note: When logging in to a local database for the first time, users are automatically prompted to connect to the Siebel Server and download the local database. Users must attempt this only after a local database has been extracted for them by the system administrator. For more information, see *Installing and Using Oracle Database SE2 for the Local Database*.

   For more information about Siebel Remote and extracting local databases, see *Siebel Remote and Replication Manager Administration Guide* .

Logging in Using the Siebel Web Client

This task is part of *Logging in to Your Siebel Application*.

You can use the Siebel Web Client to log in to the Siebel application. (In this context, *Siebel Web Client* refers to the client type that is *not* installed locally, except for the browser itself.)

To log in to your Siebel application using the Siebel Web Client

1. Open your Web browser.
2. Go to the URL for your Siebel application.
3. Log in using a valid user ID and password.

   The Siebel application that you are connecting to is already configured to work with a specific database. When using this client type, you do not specify a database to connect to.

Siebel Web Client Shortcuts and Startup Options

This topic provides information about the shortcuts installed in the Siebel program folder when you install the Siebel Web Client, and about command-line options used in these shortcuts. This topic includes the following information:

- *Siebel Web Client Shortcuts*
- *Utility and Synchronization Program Shortcuts*
- *Siebel Web Client Startup Options*
- *Creating Custom Siebel Application Shortcuts*

Siebel Web Client Shortcuts

This topic is part of *Siebel Web Client Shortcuts and Startup Options*.

The Siebel CRM installer creates shortcuts in the Siebel program folder when you install the Siebel Web Client. Separate shortcuts are installed for each installed language. Which shortcuts are installed depends on your installation choices and is also subject to the settings in the siebel.ini file at the time of installation. Most of the shortcuts are installed in the program group (program folder); some are installed in the startup group.

You can create shortcuts for any valid Siebel application for which no shortcut is created by the installer. For information about customizing the shortcuts or creating new ones, see *Siebel Web Client Startup Options* and *Creating Custom Siebel Application Shortcuts*.

Siebel application shortcuts are created by the Siebel CRM installer. When you run a shortcut to start a Siebel application using the Siebel Web Client, you can connect to the Siebel database (for LAN-connected users of the Siebel Developer Web Client), or to an initialized local database on the local computer (for Mobile Web Client). Each application shortcut loads the Siebel configuration file for that application; for example, Siebel Call Center uses the file uagent.cfg.

Utility and Synchronization Program Shortcuts

This topic is part of *Siebel Web Client Shortcuts and Startup Options*.

The following are some of the additional shortcuts created in the Siebel Web Client program folder. These shortcuts run utilities or synchronization programs.

- **Siebel Remote.** Starts the Siebel Remote stand-alone synchronization program (siebsync.exe).
- **Siebel TrickleSync.** Starts the Siebel TrickleSync program (autosync.exe). This shortcut is created in both the program group and the startup group.

Related Books

Siebel Remote and Replication Manager Administration Guide

Siebel Web Client Startup Options

This topic is part of *Siebel Web Client Shortcuts and Startup Options*.

The application shortcuts described in *Siebel Web Client Shortcuts* run an executable program called siebel.exe, followed by various startup options. For example, the shortcut for Siebel Call Center - ENU might be defined as shown in the following example (depending on your installation location). For this example, the only option used is `/c`, to specify the configuration file.

```
"C:\Siebel\Client\BIN\siebel.exe" /c CFG_FILE_PATH\uagent.cfg
```

In the preceding examples, `CFG_FILE_PATH` represents the full path to the configuration file. This file is located in `BIN\LANGUAGE` in the `SIEBEL_CLIENT_ROOT` directory. In this path, LANGUAGE is a language code, such as ENU for U.S. English.

For example, the target definition for the Siebel Call Center - ENU shortcut might resemble the following example:

```
"C:\Siebel\Client\BIN\siebel.exe" /c "C:\Siebel\Client\BIN\enu\uagent.cfg"
```

The siebel.exe program can also be run from a DOS command window. In this case, the startup options would be entered directly on the command line. For information about creating custom shortcuts, see *Creating Custom Siebel Application Shortcuts*.

The available startup options are described in the following table.

Startup Option	Description
`/c` config_file	Required. Specifies the path and file name for the configuration file to use, such as siebel.cfg for Siebel Sales or uagent.cfg for Siebel Call Center.
`/d` data_source	Specifies the data source to connect to, as defined in the configuration file: • If you do not use `/u` and `/p` to specify a valid Siebel user and do not use `/d` to specify a valid data source, then you can specify the data source from the login screen. • If you use `/u` and `/p` but do not use `/d`, then the local database is assumed.
`/l` language	Specifies the three-letter code for the language to use for this Siebel Web Client session, such as ENU for U.S. English. The applicable Siebel language must have been installed for the Siebel Web Client. If you do not use `/l` to specify a valid language, then the language is obtained from the configuration file.
`/u` username	Specifies the user name. If you do not use `/u` and `/p` to specify a valid Siebel user, then you must log in from a login screen. **CAUTION:** Consider the security issues when using `/u` and `/p` options (in particular `/p`) to access a live, production system. These values are not encrypted.
`/p` password	Specifies the password for the user specified using `/u`.
`/b` browser_exe	Specifies the path and file name for the browser executable program to use for the Siebel Web Client session. The `/b` option is used to override the default browser in Microsoft Windows.
`/s` spool_file	Specifies spooling SQL to a specified output file. This option can be useful for troubleshooting purposes. For more information, see *Siebel Performance Tuning Guide* .

Platform, database, and other certifications for Siebel are found on My Oracle Support. For help searching for certifications, see *How to Check the Certification Matrix in the New My Oracle Support Site* (KB871388).

Related Books

Siebel Security Guide

Siebel Performance Tuning Guide

Creating Custom Siebel Application Shortcuts

This task is part of *Siebel Web Client Shortcuts and Startup Options*.

If you want to customize any of the Siebel application shortcuts described in *Siebel Web Client Shortcuts*, then you can do so by modifying shortcut properties and changing the value for the Target field. Generally, rather than modifying

existing shortcuts or creating them from scratch, it is recommended that you copy existing ones and rename and modify the copies.

Observe the following requirements for creating or modifying shortcuts:

- Do not modify the location displayed in the shortcut property labeled Start in.
- When defining the Target value for a shortcut, follow the guidelines for using the options described in *Siebel Web Client Startup Options*.

To create custom Siebel application shortcuts

1. Copy existing Siebel application shortcuts.
2. Modify the copies, for example, by changing the value for the Target field.

8 Verifying and Troubleshooting Your Installation

Verifying and Troubleshooting Your Installation

This chapter provides information about verifying and troubleshooting your installation of Siebel CRM. It includes the following topics:

- *Verifying Installation for Siebel CRM*
- *Troubleshooting Installation and Configuration for Siebel CRM*
- *Verifying the ODBC Data Source*
- *Troubleshooting the ODBC Data Source Connection*
- *Verifying Installation for the Siebel Database*
- *Troubleshooting Installation for the Siebel Database*
- *Verifying Installation for Siebel Tools*
- *Verifying Installation for the Siebel Web Client*
- *Verifying Installation for Oracle Database SE2*
- *Verifying Network Connectivity for the Siebel Server Computer*
- *Verifying Network Connectivity for Mobile Users*
- *Configuring the Siebel Mobile Web Client When Neither Siebel VB nor Siebel eScript Is Licensed*

Verifying Installation for Siebel CRM

After installing Siebel CRM in a new installation, update installation, or migration installation, review the installation log files, verify the version number of your installed software, and review the directories and files that now exist under the `SIEBEL_ROOT` directory that you specified during the installation.

In addition to the steps described in this topic, perform any other tasks that might be necessary for verifying that your Siebel CRM software is functioning correctly. At the minimum, first install and configure all of the key components, including the Siebel CRM software and the Siebel database, and perform the basic verification steps for each component.

For full verification that your installed Siebel CRM server environment is working correctly, you must start one of the Siebel CRM applications in a Web browser. Start a suitable browser, specify a valid URL for one of the Siebel applications that your deployment supports, and log in using valid user credentials. Then access various screens and views in the application and make sure that the application functions properly and that the necessary seed data is present. Try this for all of the Siebel applications and languages that your deployment must support. Additional testing considerations are outside the scope of this document.

The available applications include those for which server component groups were enabled on the Siebel Server and for which Siebel Application Interface configuration was done, using Siebel Management Console. For example, the URL for Siebel Call Center might resemble the following:

https://SiebelApplicationInterfaceHost.FQDN:Port/siebel/app/callcenter/lang

In this URL:

- SiebelApplicationInterfaceHost is the host name for the installed Siebel Application Interface.
- FQDN is the fully qualified domain name for your deployment. An example is example.com.

  Note: It is strongly recommended to use the fully qualified domain name (FQDN) when you enter a URL, because security certificates are generally obtained based on fully qualified domain name.

- Port is the HTTPS redirect port number for the installed Siebel Application Interface. You specified this HTTPS redirect port during Siebel Application Interface installation.
- Lang is the deployed language for the application you are running.

This topic includes the following information:

- *Reviewing the Installation Log Files*
- *Reviewing the Log Files for Siebel Application Interface*
- *Verifying That the Application Container Is Running*
- *Verifying That the Siebel Gateway Has Started*
- *Verifying the Installation Version Number for Siebel CRM*
- *Reviewing the Installation Directories for Siebel Enterprise Components*
- *Reviewing the Installation Directories for Siebel Gateway*
- *Reviewing the Installation Directories for Siebel Server*
- *Reviewing the Installation Directories for Siebel Database Configuration Utilities*
- *Reviewing the Installation Directories for Siebel Application Interface*

Related Topics

Siebel CRM Download and Installation

Configuring Siebel CRM Server Modules

Stopping and Starting the Siebel Application Container

Customizing the Application Container for Siebel Application Interface

Reviewing the Installation Directories for Siebel Tools

Reviewing the Installation Directories for the Siebel Web Client

Troubleshooting Installation and Configuration for Siebel CRM

Reviewing the Installation Log Files

This task is part of *Verifying Installation for Siebel CRM*.

After installing the Siebel CRM modules, review the installation log files to verify that all of the modules installed successfully, particularly if you experienced any difficulties with the installation. The log files are created in the following directory:

```
SIEBEL_ROOT\cfgtoollogs\oui
```

The relevant log files for an installation session are named installActionsdate_time.log file. In the file name, date_time represents the date and time when the file was created; for example, installActions2021-01-12_10-28-04PM.log. For each set of files created, the oldest and largest log file generally represents the Siebel CRM module being installed.

Reviewing the Log Files for Siebel Application Interface

This task is part of *Verifying Installation for Siebel CRM*.

Siebel Application Interface generates one or more log files as a result of connection attempts with the Siebel Server. These log files reside in `SIEBEL_AI_ROOT\log`.

Depending on the logging level that you choose, these files record errors, warnings, and general information. Events such as invalid configuration of the Siebel Application Interface are captured in these log files. Analyzing the log files can provide clues for troubleshooting Siebel Application Interface problems.

Verifying That the Application Container Is Running

This task is part of *Verifying Installation for Siebel CRM*.

After installing one or more Siebel CRM modules using the Siebel CRM installer, verify that the application container is running. You can also check relevant log files, such as catalina.log, that are created in the following directories:

- `SIEBEL_ROOT\applicationcontainer_internal\logs` (for Siebel Enterprise components)
- `SIEBEL_ROOT\applicationcontainer_external\logs` (for Siebel Application Interface)

Also verify information about the application container stored in subdirectories. For example, the server.xml file, located in the `conf` subdirectory, can be checked to validate the HTTPS port number and other information for the installed application container.

Verifying That the Siebel Gateway Has Started

This task is part of *Verifying Installation for Siebel CRM*.

You must make sure that the Siebel Gateway is started when you do any of the following:

- Configure the Security profile
- Create and deploy the Siebel Enterprise profile
- Create and deploy a Siebel Server profile
- Create and deploy a Siebel Application Interface profile
- Operate any of the Siebel applications
- Remove the configuration of a Siebel Application Interface, Siebel Server or the Siebel Enterprise (tasks that you might perform when you are uninstalling the software)

Note: It is recommended that you review the Siebel Gateway log files, which are found in the `SIEBEL_GATEWAY_ROOT\LOG` directory.

To start the Siebel Gateway manually on Windows

1. Navigate to Control Panel, Administrative Tools, and then Services.
2. If the Siebel Gateway is not started, then click Action, and then Start.

Verifying the Installation Version Number for Siebel CRM

This task is part of *Verifying Installation for Siebel CRM*.

After installing the Siebel CRM modules, verify the version number of the installed software.

To verify the installation version number for Siebel CRM

1. Navigate to the installation directory for your Siebel CRM release. For example:

   ```
   C:\Siebel_SES
   ```

2. Open the file Siebel_version.properties in a text editor.

   The file content is displayed, resembling the following example:

   ```
   #Fri July 22 18:21:52 PST 2022
   SIEBEL_VERSION=22.7.0.0.0
   ```

Reviewing the Installation Directories for Siebel Enterprise Components

This task is part of *Verifying Installation for Siebel CRM*.

After installing Siebel CRM, review the directory structure created by the installer for the Siebel Enterprise Components. For example, some or all of the directories described in the following table might now exist in the installation directory. Which directories are present in the installation directory depends on which Siebel CRM modules you installed, whether it was a new installation or an update or migration installation, whether you configured the application containers, and whether you completed configuration in Siebel Management Console.

> **Note:** You can optionally install Siebel Web Client or Siebel Tools with Siebel Enterprise Components in a nonproduction environment. If you do so, then some of the directories listed here, such as `cfgtoollogs`, are also relevant to your Siebel Tools or Siebel Web Client installation. However, several additional files and directories specific to the installed client modules are also created in the top-level installation directory alongside of those for the server modules. For additional directories that might have been created, see also *Reviewing the Installation Directories for Siebel Tools* and *Reviewing the Installation Directories for the Siebel Web Client*.

Directory	Description
SIEBEL_ROOT	**SIEBEL_ROOT.** Top-level directory for your Siebel CRM installation. The actual directory name is the directory where you installed the software.

Directory	Description
SIEBEL_ROOT\applicationcontainer_external	**applicationcontainer_external.** Directory used by the application container for Siebel Application Interface. Several of these files are updated by the installer at the end of the installation process and can be examined to help make sure that installation was successful.
SIEBEL_ROOT\applicationcontainer_external\conf	**conf.** Contains the file server.xml, which can be examined for information such as the HTTPS port number of the installed application container.
SIEBEL_ROOT\applicationcontainer_external\webapps	**webapps.** Contains files for the Web applications supported by the application container, including properties files. The subdirectory **`siebel`** contains the **`WEB-INF`** subdirectory, which contains files such as web.xml and log4j2-siebel.xml.
SIEBEL_ROOT\applicationcontainer_external\siebwebroot	**siebelwebroot.** Contains the default HTML file (default.htm) and Web artifact subdirectories, including **`files`**, **`fonts`**, **`htmltemplates`**, **`images`**, **`migration`**, **`scripts`**, and **`smc`**. The **`files`** directory contains CSS files.
SIEBEL_ROOT\applicationcontainer_internal	**applicationcontainer_internal.** Directory used by the application container for Siebel Enterprise components (Siebel Server, Siebel Gateway, and so on). Several of these files are updated by the installer at the end of the installation process and can be examined to help make sure that installation was successful.
SIEBEL_ROOT\applicationcontainer_internal\conf	**conf.** Contains the file server.xml, which can be examined for information such as the HTTPS port number of the installed application container.
SIEBEL_ROOT\applicationcontainer_internal\webapps	**webapps.** Contains files for the Web applications supported by the application container, including properties files. The subdirectory **`siebel`** contains the **`WEB-INF`** subdirectory, which contains files such as web.xml and log4j2-siebel.xml.
SIEBEL_ROOT\applicationinterface	**applicationinterface.** Directory used by Siebel Application Interface.
SIEBEL_ROOT\cfgtoollogs	**cfgtoollogs.** Directory used for installation log files.
SIEBEL_ROOT\config	**config.** Directory used for configuraiton files.
SIEBEL_ROOT\dbsrvr	**dbsrvr.** Directory used by Database Configuration Utilities.
SIEBEL_ROOT\diagnostics	**diagnostics.**
SIEBEL_ROOT\eaiconn	**eaiconn.** Directory used by EAI connector.
SIEBEL_ROOT\gtwysrvr	**gtwysrvr.** Directory used by Siebel Gateway.
SIEBEL_ROOT\inventory	**inventory.** Directory used for the installation inventory.
SIEBEL_ROOT\jre	**jre.** Directory used by the Java Runtime Environment.

Directory	Description
SIEBEL_ROOT\manifest	**manifest.** Directory used for manifest files.
SIEBEL_ROOT\oui	**oui.** Directory used by Oracle Universal Installer software.
SIEBEL_ROOT\siebsrvr	**siebsrvr.** Directory used by Siebel Server.

Reviewing the Installation Directories for Siebel Gateway

This task is part of *Verifying Installation for Siebel CRM*.

After installing Siebel Enterprise Components, review the directory structure created by the installer. Verify that the following directories and files used by Siebel Gateway, and described in the following table, now exist in the installation directory.

Note: Siebel Gateway is no longer a selectable option for new installations and is always installed where you install Siebel Enterprise Components.

Directory	Description
gtwysrvr	**gtwysrvr.** Top-level directory for Siebel Gateway (under `SIEBEL_ROOT`).
\ADMIN	**ADMIN.** The template files used for scripts that control the running and configuration of the Siebel Gateway.
\bin	**bin.** A symbolic link to `SIEBEL_ROOT\siebsrvr\bin`, which contains the Siebel Gateway start script, stop script, and executable programs.
\classes	**classes.** Contains files used in the configuration process.
\EXTCACHE	**EXTCACHE.**
\lib	**lib.** A symbolic link to `SIEBEL_ROOT/siebsrvr/lib`, which contains the Siebel Gateway start script, stop script, and executable programs.
\locale	**locale.** A symbolic link to `SIEBEL_ROOT\siebsrvr\locale`, which contains language-specific files and scripts. These are not configurable.
\log	**log.** Siebel Gateway log files.
\registry	**registry.** Contains files used by Apache ZooKeeper, which maintains the Siebel Gateway registry. (Prior to Siebel CRM 21.4, the `registry` directory was named `zookeeper`.)

Directory	Description
\upgrade.log	**upgrade.log.** File that contains information logged during an upgrade.

Reviewing the Installation Directories for Siebel Server

This task is part of *Verifying Installation for Siebel CRM*.

After installing Siebel Enterprise Components, review the directory structure created by the installer. Verify that the following directories and files used by Siebel Server, and described in the following table, now exist in the installation directory.

Note: Siebel Server is no longer a selectable option for new installations and is always installed where you install Siebel Enterprise Components.

Directory	Description
siebsrvr	**siebsrvr.** Top-level directory for Siebel Server (under `SIEBEL_ROOT`).
\ADMIN	**ADMIN.** The template files used for scripts that control the running and configuration of Siebel Server.
\bin	**bin.** Contains binary library files and executable programs or scripts (Windows only) for Siebel Enterprise Server components, as well as DSN files, SQL files, user preference files, and configuration files (located in language subdirectories) and language subdirectories related to language-specific server components.
\CLASSES	**CLASSES.** Contains files used in the configuration process.
\DBTEMPL	**DBTEMPL.** Contains dictionary and local database files required by Siebel Remote for regional users and Mobile Web Client users.
\DOCKING	**DOCKING.** Contains transaction files, visibility, and other databases required by Siebel Remote.
\ESCRIPTTMPL	**ESCRIPTTMPL.**
\HELP	**HELP.** Contains help files.
\INPUT	**INPUT.** Contains files related to Siebel Remote.
\ISSTEMPL	**ISSTEMPL.**
\jre	**jre.** Contains files for Java Runtime Environment.

Directory	Description
\JSON	**JSON.** Contains JSON (JavaScript Object Notation) files.
\LEX	**LEX.** Language-related files.
\lib	**lib.** (UNIX only) Contains the binary library files and executable programs or scripts.
\LOCALE	**LOCALE.** Contains language-specific files. These files are not configurable.
\log	**log.** Contains client and utility log files.
\MSGTEMPL	**MSGTEMPL.** Stores language-specific files for mail merge.
\output	**output.** Contains files related to Siebel Remote.
\plugins	**plugins.**
\REPORTS	**REPORTS.** Contains the report executable programs used by Siebel Proposals to include reports in proposals.
\SDQConnector	**SDQConnector.** Contains the DLLs, configuration files, and other files necessary to connect the Siebel Data Quality Universal Connector to one or more external data quality products. An external data quality product is validated though Oracle's partner initiative.
\SEARCH	**SEARCH.** Contains the indexes and scripts used to administer and execute searches.
\SMARTANSWER	**SMARTANSWER.** Contains files related to Siebel Smart Answer.
\SQLTEMPL	**SQLTEMPL.** Contains SQL statements used by Siebel Server components. Do not modify these files.
\TEMP	**TEMP.** Stores temporary files for use by the Siebel Server.
\UPGRADE	**UPGRADE.** Contains files and scripts related to version upgrades of Siebel CRM. Also holds temporary, backup, and state log files used during an upgrade.
\upgrade.log	**upgrade.log.** File that contains information logged during an upgrade.
\XML	**XML.**
\XMLP	**XMLP.**
\xsd	**xsd.**
\XSLT	**XSLT.**

Directory	Description

Reviewing the Installation Directories for Siebel Database Configuration Utilities

This task is part of *Verifying Installation for Siebel CRM*.

After installing Siebel Enterprise Components, review the directory structure created by the installer. Verify that the following directories and files used by Database Configuration Utilities, and described in the following table, now exist in the installation directory.

Note: Database Configuration Utilities is no longer a selectable option for new installations and is always installed where you install Siebel Enterprise Components. As before, you would typically use the Database Configuration Utilities content that was installed with the first Siebel Server.

Directory	Description
dbsrvr	**dbsrvr.** Top-level directory for Siebel Database Configuration Utilities (under `SIEBEL_ROOT`).
\bin	**bin.** A symbolic link to `SIEBEL_ROOT\siebsrvr\bin`, which contains executable files.
\COMMON	**COMMON.** Contains database platform-independent files.
\DBSRVR_ROOT	**DBSRVR_ROOT (ORACLE, DB2UDB, DB2390, or MSSQL).** Contains scripts specific to your database, including upgrade scripts for previous versions of Siebel CRM.
\DBSRVR_ROOT\DBOUTPUT	**DBOUTPUT (IBM DB2 for z/OS only).** Contains subdirectories in which DDL is deposited by the installer for later transfer to the DB2 host when the Siebel Schema installation option Generate DDL Into Files is chosen.
\DBSRVR_ROOT\SIEBPROC	**SIEBPROC (IBM DB2 only).** Contains User Defined Functions (UDFs) and stored procedures for IBM DB2, by operating system. See also *Installing the Stored Procedures and User-Defined Functions on IBM DB2*.
\DBSRVR_ROOT\SIEBPROC\AIX	**AIX.** UDFs and stored procedures for IBM DB2 on AIX.
\DBSRVR_ROOT\SIEBPROC\HPUX	**HPUX.** UDFs and stored procedures for IBM DB2 on HP-UX.
\DBSRVR_ROOT\SIEBPROC\LINUX	**LINUX.** UDFs and stored procedures for IBM DB2 on Linux.
\DBSRVR_ROOT\SIEBPROC\SOLARIS	**SOLARIS.** UDFs and stored procedures for IBM DB2 on Oracle Solaris.

Directory	Description
\DBSRVR_ROOT\SIEBPROC\WIN32	**WIN32.** UDFs and stored procedures for IBM DB2 on Windows.
\DBSRVR_ROOT\SQLPROC	**SQLPROC (IBM DB2 only).** Contains subdirectories that contain the binary files that are required to install stored procedures. These stored procedures perform data migration as part of an upgrade.
\DBSRVR_ROOT\SQLPROC\AIX	**AIX.** Stored procedures for IBM DB2 on AIX.
\DBSRVR_ROOT\SQLPROC\HPUX	**HPUX.** Stored procedures for IBM DB2 on HP-UX.
\DBSRVR_ROOT\SQLPROC\LINUX	**LINUX.** Stored procedures for IBM DB2 on Linux.
\DBSRVR_ROOT\SQLPROC\SOLARIS	**SOLARIS.** Stored procedures for IBM DB2 on Oracle Solaris.
\DBSRVR_ROOT\SQLPROC\WIN32	**WIN32.** Stored procedures for IBM DB2 on Windows.
\DBSRVR_ROOT\STORPROC	**STORPROC (IBM DB2 for z/OS only).** Contains stored procedures for DB2 for z/OS.
\DBSRVR_ROOT\UPGRADE	**UPGRADE.** Directories containing files to enable upgrading from specific versions of Siebel CRM that are supported for upgrade to the current release.
\FILES	**FILES.** This directory contains sample file attachments. You must copy these files to the appropriate subdirectory of the Siebel File System. See *Populating the Siebel File System*.
\LANG	**LANG.** Contains language-specific and database-specific files for the ancestor Siebel Repository and supporting files. For example, ENU contains language-specific files for U.S. English, and DEU contains language-specific files for German.
\lib	**lib.** (UNIX only) A symbolic link to `SIEBEL_ROOT/siebsrvr/bin`, which contains executable files.
\LOCALE	**LOCALE.** Contains translation files (for Oracle use only).

Reviewing the Installation Directories for Siebel Application Interface

This task is part of *Verifying Installation for Siebel CRM*.

After installing Siebel Enterprise Components, review the directory structure created by the installer. Verify that the following directories and files used by Siebel Application Interface, and described in the following table, now exist in the installation directory. See also the information about the `applicationcontainer_external` directory in *Reviewing the Installation Directories for Siebel Enterprise Components*.

Note: Siebel Application Interface is no longer a selectable option for new installations and is always installed where you install Siebel Enterprise Components.

Directory	Description
applicationinterface	Top-level directory for Siebel Application Interface (under `SIEBEL_ROOT`).

Troubleshooting Installation and Configuration for Siebel CRM

This topic presents troubleshooting information related to installation and configuration of Siebel CRM server modules. Some of the information also applies to additional modules that use the same installer and are also configured using Siebel Management Console.

This topic includes the following information:

- *Troubleshooting Installation and Configuration for Siebel Gateway*
- *Troubleshooting Installation and Configuration for Siebel Server*
- *Troubleshooting Installation and Configuration for Siebel Application Interface*

Related Topics

Starting the Siebel Server Service on Windows

Verifying Installation for Siebel CRM

Stopping and Starting the Siebel Application Container

Troubleshooting Installation and Configuration for Siebel Gateway

This task is part of *Troubleshooting Installation and Configuration for Siebel CRM*.

This topic describes potential errors that can result from a faulty installation or configuration of Siebel Gateway. Such problems can have any of several causes, some of the most common of which are listed in the following table.

Note: If you cannot start the Siebel Gateway, then you will not be able to configure a Siebel Enterprise or configure and start a Siebel Server.

Problem	Cause	Solution
The Siebel Gateway does not start.	You might not have privileges as the Siebel service owner.	Review the instructions in *Creating the Siebel Service Owner Account*.

Problem	Cause	Solution
You cannot start the Siebel Gateway service on \ \COMPUTER_NAME. `Error XXX: The service did not start due to logon failure.`	Login failure	1. Navigate to Control Panel, Administrative Tools, and then Services. 2. Select the Siebel Gateway. 3. Right-click, then choose Properties. 4. In the Log On tab, specify the user name and password to use for this service.
You receive an error at system startup that a driver is missing.	This problem can occur if you uninstall earlier Siebel CRM versions without first stopping the Siebel Server.	Stop the Siebel Server before uninstalling Siebel CRM. If the services are no longer visible within the Services menu, then contact Oracle Global Customer Support for instructions.

Troubleshooting Installation and Configuration for Siebel Server

This task is part of *Troubleshooting Installation and Configuration for Siebel CRM*.

This topic describes potential errors that can result from a faulty installation or configuration of Siebel Server. Such problems can have any of several causes, some of the most common of which are listed in the following table.

Problem	Cause	Solution
The Siebel Server cannot be installed.	Insufficient user privileges	For information about setting up appropriate administrative user privileges to install, see .
	Trying to install or configure the Siebel Server out of sequence	For the required installation and configuration sequence, see .
	Failure to install the required hardware or software	Installation errors related to software requirements are logged in the Siebel CRM installer log file.
The Siebel Server does not start after configuration.	Siebel Gateway not started	Verify that the Siebel Gateway was started. Start it if it was stopped.
	Invalid input values for Siebel Server profile creation and deployment	Verify that the input values were valid.
	Insufficient system privileges	Verify that you have sufficient system privileges to start the service. For more information, see .
Cannot save a new profile in Siebel Management Console.	The specified profile name might already exist.	Provide a unique name for the new profile and retry saving the profile.

Problem	Cause	Solution
ODBC connection failure	Failure to configure ODBC connections (Failed connectors are displayed during confiuguration)	ODBC connections can be set up manually in case of failed automated configuration.
URL does not bring up Siebel Management Console.	Installations might not have been correctly performed. Security certificates might not be valid, as configured. You might have specified an incorrect port number in the URL. Other causes might apply.	Verify the following: • The installations were correctly performed, • Valid security certificates were used, and the security authentication system is available • The Siebel Application Interface application container is running • The URL is correct and includes the correct HTTP or HTTPS port See also *Troubleshooting Installation and Configuration for Siebel Application Interface*. For security issues, see also *Siebel Security Guide* .
URL does not bring up login page for a Siebel application. Cannot log in to a Siebel application.	Configuration of Siebel Server or Siebel Application Interface might not have been correctly performed or completed. Siebel database information might have been specified incorrectly. The Application Object Manager might not have been enabled on the Siebel Server. You might have specified an incorrect port number in the URL. You might have added components on a new Siebel Server, but neglected to update the Siebel Application Interface profile. Siebel system services might not be running. You might be experiencing problems with the security authentication system you are using. Other causes might apply.	Verify the following: • The installations and configurations were correctly performed, and database information was specified correctly • Security authentication was correctly configured and the security authentication system is available • The Siebel Application Interface application container is running • The Siebel Gateway and Siebel Server services are running • The URL is correct and includes the correct HTTP or HTTPS port See also *Troubleshooting Installation and Configuration for Siebel Application Interface*. For security issues, see also *Siebel Security Guide* .

Platform, database, and other certifications for Siebel are found on My Oracle Support. For help searching for certifications, see *How to Check the Certification Matrix in the New My Oracle Support Site* (KB871388).

Troubleshooting Installation and Configuration for Siebel Application Interface

This task is part of *Troubleshooting Installation and Configuration for Siebel CRM*.

This topic provides suggestions for troubleshooting problems that you might encounter when installing and configuring Siebel Application Interface. Typical problems are shown in the following table.

Problem	Cause	Solution
Siebel Application Interface profile deployment fails despite no obviously incorrect profile values.	The application container might not have been available to support the configuration process.	Try the following: 1. Clear the CGHostURI entry in the applicationinterface.properties file, in `SIEBEL_AI_ROOT\applicationcontainer_external\webapps`. 2. Restart the application container, as described in *Stopping and Starting the Siebel Application Container*. 3. Deploy Siebel Application Interface again.
After installation, when the Siebel Web Client is started, a message appears, stating: `Page Cannot be displayed`	The application container is not running.	Make sure that the application container is running. Refresh the connection between your browser and the Siebel Application Interface.
	The Siebel Application Interface port is incorrectly specified.	Verify that the Siebel Application Interface port information is correct.
	Applications were not configured properly.	Make sure that the Local Path for the configured applications is correct and resembles the following: `SIEBEL_AI_ROOT\applicationcontainer_external\webapps\siebel`
	Anonymous users have incorrect responsibilities.	Make sure that the specified anonymous users are also defined in the Siebel database with the correct responsibilities. Otherwise, end users cannot access the home page.
	The connect string for the Siebel application is incorrect.	Make sure that the connect string for the Siebel application is correct. The value resembles the following example: `ConnectString = siebel.TCPIP.none.none://SiebelServerHost:2321/EnterpriseServerName/ProductNameObjMgr_language`
	Siebel Server components or component groups might not be enabled.	Make sure that the necessary Siebel Server components and component groups are enabled.
Your Siebel application stops responding, displays	Appropriate settings are not available within the application	Make sure that the configuration contains valid anonymous user and port number values.

Problem	Cause	Solution
a blank page or page not found, or times out.	configuration defined in the profile for the Siebel Application Interface.	Also make sure that appropriate values were specified for session time-out and guest session timeout. For information about configuring these and other settings, see *Creating a Siebel Application Interface Profile*.
Inability to access the Siebel Web Client. The browser status bar might display errors such as: `SWESubmitOnEnter is undefined` Also, the login page might stop responding. Or, the Siebel Web Client login page does not display properly; for example, images might be missing.	The user account running the Siebel Application Interface does not have proper permissions to the `SIEBEL_AI_ROOT\applicationcontainer_external\webapps\siebel` directory.	Stop the Siebel Application Interface. Make sure that the permissions meet the requirements described in *Requirements for Installing and Configuring the Siebel Application Interface*. Restart the Siebel Application Interface, as described in *Stopping and Starting the Siebel Application Container*.

Verifying the ODBC Data Source

This topic provides information about verifying the ODBC data source for your enterprise database platform. It includes the following information:

- *Verifying the ODBC Data Source for Oracle Database*
- *Verifying the ODBC Data Source for IBM DB2*
- *Verifying the ODBC Data Source for Microsoft SQL Server and Azure SQL Database Server*

Related Topics

Troubleshooting the ODBC Data Source Connection

Planning RDBMS Installation and Configuration

Verifying the ODBC Data Source for Oracle Database

This task is part of *Verifying the ODBC Data Source*.

Configuring the Siebel Enterprise automatically creates an ODBC system data source name (DSN) that the Siebel Server uses to connect to the Siebel database on the RDBMS.

Make sure that a supported version of the Oracle Database Client has been installed and that the Siebel Server service is started prior to verifying the ODBC data source. This topic applies also to virtual ODBC data sources in a clustered environment.

CAUTION: In general, do not change the default settings that are created automatically with the ODBC data source. However, if you have upgraded from an Oracle Database that does not use CBO, or if you manually created your ODBC, then you must check your ODBC settings, as shown in the following procedure.

To verify the ODBC data source for Oracle Database (on Windows)

1. Start the Microsoft ODBC Administrator.

   Run the 32-bit version of the Microsoft ODBC Administrator, which is located here: `c:\Windows\SysWOW64\odbcad32.exe`.

   Note: You need the 32-bit version of the Microsoft ODBC Administrator because Siebel CRM uses 32-bit ODBC drivers and database clients only.

2. On the ODBC Data Source Administrator dialog box, select the System DSN tab.
3. Review the data source name; its default name is SiebelEnterpriseName_DSN.

   In the data source name, SiebelEnterpriseName is the name that you gave the Siebel Enterprise during its configuration.
4. Record the name of the ODBC data source in *Siebel Deployment Planning Worksheet* if you have not already done so.
5. Select the data source SiebelEnterpriseName_DSN, and click Configure.

   The ODBC Oracle Driver Setup screen appears.
6. To test the connection, click Test Connect.

   If the connection is valid, then you see a message box confirming the connection.

   If the connection could not be made, then see *Verifying Network Connectivity for the Siebel Server Computer* and *Troubleshooting the ODBC Data Source Connection*.
7. In the Windows registry, locate the following registry entry:

   ```
   [HKEY_LOCAL_MACHINE\SOFTWARE\ODBC\ODBC.INI\SiebelEnterpriseName_DSN]
   ```

8. Verify that the following registry keys are set as shown. These values are required in order for the ODBC driver to behave correctly.

   ```
   LobPrefetchSize=8388608
   FetchBufferSize=1048576
   ```

Verifying the ODBC Data Source for IBM DB2

This task is part of *Verifying the ODBC Data Source*.

Configuring the Siebel Enterprise automatically creates an ODBC system data source name (DSN) that the Siebel Server uses to connect to the Siebel database on the RDBMS.

Make sure that the appropriate database client has been installed and that the Siebel Server service is started prior to verifying the ODBC data source. This topic applies also to virtual ODBC data sources in a clustered environment.

CAUTION: Do not change the default settings that are created automatically with the ODBC data source.

Note: For information about verifying the ODBC data source for IBM DB2 for z/OS, see *Implementing Siebel Business Applications on DB2 for z/OS* .

To verify the ODBC data source for IBM DB2 (on Windows)

1. Start the Microsoft ODBC Administrator.

   Run the 32-bit version of the Microsoft ODBC Administrator, which is located here: `c:\Windows\SysWOW64\odbcad32.exe`.

   Note: You need the 32-bit version of the Microsoft ODBC Administrator because Siebel CRM uses 32-bit ODBC drivers and database clients only.

2. On the ODBC Data Source Administrator dialog box, select the System DSN tab.
3. Review the data source name; its default name is SiebelEnterpriseName_DSN.

   In the data source name, SiebelEnterpriseName is the name that you gave the Siebel Enterprise during its configuration.
4. Record the name of the ODBC data source in *Siebel Deployment Planning Worksheet* if you have not already done so.
5. Select the data source SiebelEnterpriseName_DSN, and click Configure.

   A DB2 Message box appears to let you know your connection status.

   If you are not connected, then you are prompted whether you want to connect now to the data source.
6. Click Yes.

   The Connect to DB2 Database screen appears.
7. Type your user ID and passwords into the applicable fields, and click OK.

   If the connection is valid, then you see a message box confirming the connection.

   If the connection could not be made, then see *Verifying Network Connectivity for the Siebel Server Computer* and *Troubleshooting the ODBC Data Source Connection*.
8. From the Windows command prompt, navigate to `SIEBEL_ROOT\siebsrvr\BIN`, and execute the following command:

   ```
   odbcsql /s ODBC_DSN /u database_account_name /p password
   ```

   where:

   - ODBC_DSN is the ODBC data source name, in the form SiebelEnterpriseName_DSN. In the data source name, SiebelEnterpriseName is the name of your Siebel Enterprise.
   - database_account_name is a valid database account name.
   - password is the corresponding database account password.

   When you connect, a confirmation message appears.

Verifying the ODBC Data Source for Microsoft SQL Server

This task is part of *Verifying the ODBC Data Source*.

Configuring the Siebel Enterprise automatically creates an ODBC system data source name (DSN) that the Siebel Server uses to connect to the Siebel database on the RDBMS.

Make sure that the appropriate database client has been installed and that the Siebel Server service is started prior to verifying the ODBC data source. This topic applies also to virtual ODBC data sources in a clustered environment.

CAUTION: Do not change the default settings that are created automatically with the ODBC data source.

To verify the ODBC data source for Microsoft SQL Server and Azure SQL Database Server

1. Start the Microsoft ODBC Administrator.

   Run the 32-bit version of the Microsoft ODBC Administrator, which is located here: `c:\Windows\SysWOW64\odbcad32.exe`.

   Note: You need the 32-bit version of the Microsoft ODBC Administrator because Siebel CRM uses 32-bit ODBC drivers and database clients only.

2. On the ODBC Data Source Administrator dialog box, select the System DSN tab.
3. Review the data source name; its default name is SiebelEnterpriseName_DSN.

   In the data source name, SiebelEnterpriseName is the name that you gave the Siebel Enterprise during its configuration.
4. Record the name of the ODBC data source in *Siebel Deployment Planning Worksheet* if you have not already done so.
5. Select the data source SiebelEnterpriseName_DSN, and click Configure.

   The Microsoft SQL Server DSN Configuration screen appears.
6. Click Next, and click Client Configuration.

   The Edit Network Library Configuration screen appears.
7. Verify that the port number for the database is correct.
8. If the port number is incorrect, then edit it, and then click OK.

   To edit the port number, you might have to first select and then deselect the check box Dynamically Determine Port.

   The Microsoft SQL Server DSN Configuration screen appears.

9. Select verification with SQL Server using a valid database login ID and password that you enter in the Login ID and Password fields. Then click Next.

   The program tests the connection:

   - If the connection is valid, then you see a message box confirming the connection.
   - If the connection could not be made, then see *Verifying Network Connectivity for the Siebel Server Computer* and *Troubleshooting the ODBC Data Source Connection*.

Troubleshooting the ODBC Data Source Connection

This topic describes troubleshooting steps for your ODBC data source connection.

Note: If your database is IBM DB2 for z/OS, then see *Implementing Siebel Business Applications on DB2 for z/OS* .

This topic includes the following information:

- *Troubleshooting Connection Problems on Oracle Database*
- *Troubleshooting Connection Problems on IBM DB2*
- *Troubleshooting Connection Problems on Microsoft SQL Server and Azure SQL Database Server*

Troubleshooting Connection Problems on Oracle Database

This task is part of *Troubleshooting the ODBC Data Source Connection*.

Complete the following instructions to troubleshoot Oracle Database connection problems on Windows. See also *Configuring Siebel Server Connectivity to the Siebel Database*.

To troubleshoot a failed ODBC connection for Oracle Database (on Windows)

1. Verify that the ODBC driver was correctly installed by reviewing the file structure under `SIEBSRVR_ROOT\BIN`.
2. If the files have been correctly copied to the `BIN` subdirectory, then verify that the Oracle connect string that you entered during the Siebel Enterprise configuration was valid.

Troubleshooting Connection Problems on IBM DB2

This task is part of *Troubleshooting the ODBC Data Source Connection*.

Complete the following instructions to troubleshoot IBM DB2 connection problems. See also *Configuring Siebel Server Connectivity to the Siebel Database*.

To troubleshoot a failed ODBC connection for IBM DB2

- Verify that the data source name (SiebelEnterpriseName_DSN) that you entered during the Siebel Server configuration was valid. Also verify that the required IBM DB2 client version is available on the computer and verify that the IBM DB2 catalog step has been completed correctly.

Troubleshooting Connection Problems on Microsoft SQL Server

This task is part of *Troubleshooting the ODBC Data Source Connection*.

Complete the following instructions to troubleshoot Microsoft SQL Server connection problems on Windows. See also *Configuring Siebel Server Connectivity to the Siebel Database*.

To troubleshoot a failed ODBC connection for Microsoft SQL Server and Azure SQL Database Server

- Verify that the data source name (SiebelEnterpriseName_DSN) that you entered during the Siebel Server configuration was valid. Also verify that the SQL Server database is up and running, and validate the ODBC connectivity.

Verifying Installation for the Siebel Database

After installing the Siebel database on the RDBMS using the Siebel Database Configuration Wizard, review the installation log files and the system preferences for the database code page. This topic includes the following information:

- *Reviewing the Installation Log Files for the Siebel Database*
- *Verifying System Preferences and Settings for the Database Code Page*

Related Topics

Installing the Siebel Database

Verifying Installation for Siebel CRM

Troubleshooting Installation for the Siebel Database

Reviewing the Installation Log Files for the Siebel Database

This task is part of *Verifying Installation for the Siebel Database*.

Installing Siebel database components on the RDBMS creates several log files within the `SIEBEL_ROOT\log` subdirectory. You must review the files in this subdirectory for any errors.

(Windows) For example, log files created might be named upgwiz.log (the most recently created), upgwiz_01.log, upgwiz_02.log, and so.

(UNIX) For example, log files created might be named srvrupgwiz1.log (the most recently created), srvrupgwiz1_01.log, srvrupgwiz1_02.log, and so.

> **Note:** For information about reviewing log files using the `logparse` utility, see *Siebel Database Upgrade Guide* . This utility is located in `SIEBSRVR_ROOT\bin`.

Some other files generated, by supported RDBMS, are listed as follows.

Oracle Database. For Oracle Database, the following log files are also generated when you create tables, indexes, and seed data:

```
ddl_ctl.log
dataimp_prim.log
dataimp_prim_lang.log
ddlseq.log
ddlora.log
ddlview.log
ddlview_sia.log
seedssa.log
seedver.log
pkgseq.log
pkgldel.log
seedora.log
pkgvis.log
trgreset.log
ifstrg.log
ifindxstrg.log
set_utc_on.log
month_fn.log
```

IBM DB2. For IBM DB2, the following log files are also generated when you create tables, indexes, and seed data:

```
ddl_ctl.log
dataimp_prim.log
dataimp_prim_lang.log
db2ddl.log
siebproc.log
seeddb2.log
seedver.log
seedssa.log
ddlview.log
ddlview_sia.log
grantstat.log
updatestats.log
loadstats.log
set_utc_on.log
```

Microsoft SQL Server/Azure SQL Database Server. The following log files are also generated when you create tables, indexes, and seed data:

```
helpsort.log
ddl_ctl.log
dataimp_prim.log
dataimp_prim_lang.log
pkgseq.log
seedmss.log
ddlms.log
trgreset.log
seedver.log
seedssa.log
ddlview.log
ddlview_sia.log
set_utc_on.log
```

Verifying System Preferences and Settings for the Database Code Page

This task is part of *Verifying Installation for the Siebel Database*.

After you complete installing Database Configuration Utilities and installing the Siebel database, you must verify system preferences and other settings for the Siebel application that indicate whether you are using Unicode, and indicate the code page that you are using.

For more information about code pages, see *Planning RDBMS Installation and Configuration*.

The following procedure assumes that you have already installed Siebel Tools.

- For information about installing Siebel Tools, see *About Installing Siebel Web Client or Siebel Tools*.
- For information about starting Siebel Tools, see *Verifying Successful Installation of Siebel Tools*.

Platform, database, and other certifications for Siebel are found on My Oracle Support. For help searching for certifications, see *How to Check the Certification Matrix in the New My Oracle Support Site* (KB871388).

To verify system preferences and settings for the database code page

1. Start Siebel Tools and connect to the Siebel database.
2. Navigate to Screens, System Administration, and then System Preferences.
3. Look for `System Preference Name = Enterprise DB Server Code Page`. Verify that the value has been set correctly, based on the value that you selected during the installation of the Siebel database components (see *Installing the Siebel Database*).

   > **Note:** The value of Enterprise DB Server Code Page must be in lowercase, for example, utf-8 or utf-16. An Oracle Database character set might have a different value, although the system preference value must be utf-8. The utf-16 code page is also known as UCS-2, although the system preference value must be utf-16.

   Possible values for the system preference are listed in the following table:

Language	Database	Value
All	Oracle Database	utf-8 (for Unicode)
All	IBM DB2, Microsoft SQL Server/ Azure SQL Database Server	utf-16 (for Unicode)
Japanese	Oracle Database, IBM DB2	cp932 (or equivalent)
Western European	All	cp1252 (or equivalent)

4. Verify that the column UNICD_DATATYPS_FLG in the table S_APP_VER is set correctly. The flag value must be in uppercase.

Possible values for UNICD_DATATYPS_FLG are listed in the following table:

Code Page	Database	Value
Non-Unicode code page	All	N
Unicode	Oracle Database	8
Unicode	IBM DB2	Y
Unicode	Microsoft SQL Server/Azure SQL Database Server	Y

To specify that you are using an EBCDIC code page

If you are using an EBCDIC code page for DB2 on z/OS, then you must set the EBCDICCodePage parameter to specify that you are using this type of code page. This advanced parameter replaces the system preference of the same name. EBCDICCodePage is now a component level parameter and is recommended to be set at the component level.

The EBCDICCodePage parameter changes the sort order to ASCII in your EBCDIC database, which prevents intermittent workspace delivery failures related to the sort order of WS_SRC_ID records.

To set this parameter at the component level (such as an Application Object Manager component), run Server Manager and enter a command like the following:

```
change param EBCDICCodePage=TRUE for comp component_name
```

To set this parameter at the enterprise level, run Server Manager and enter a command like the following:

```
change entparam EBCDICCodePage=TRUE
```

If you are using Siebel Tools, then set this parameter in the [Siebel] section of the tools.cfg file, as follows:

```
[Siebel]
EBCDICCodePage=TRUE
```

Troubleshooting Installation for the Siebel Database

These topics provide suggestions for troubleshooting problems that you might encounter when installing and configuring the Siebel database.

Typically, problems during database installation result from insufficient storage space having been allocated, or from the installer having improper user privileges.

This topic includes the following information:

- *Acceptable Errors for Installation of the Siebel Database*

- *Troubleshooting Siebel Repository Import for the Siebel Database*

Related Topics

Installing the Siebel Database

Verifying Installation for Siebel CRM

Verifying Installation for the Siebel Database

Acceptable Errors for Installation of the Siebel Database

This task is part of *Troubleshooting Installation for the Siebel Database*.

The log files might include errors that are expected and benign. Compare any error messages found in the log files to the sample error messages in the errors.rtf file, which is located in the installation subdirectory for your database, for example, ORACLE or (Windows) MSSQL. If a log file is not listed in the errors.rtf file, then there are no acceptable error messages for that log file. No further action is required if the log files contain errors listed in the errors.rtf file.

Note: Only one of each type of error occurring in a particular log file appears in the errors.rtf file.

If you find errors that are not listed in the errors.rtf file, then correct the condition that caused the errors, and rerun the Upgrade Wizard. The wizard restarts from the point where it left off.

Do not review only the error numbers because these might have changed following installation of a new driver version. Instead, compare the error descriptions to find out which are acceptable errors for this database.

CAUTION: Although other errors are rarely encountered, this review is critical. Certain errors, such as a failure to create indexes, might result in performance problems or anomalous behavior in Siebel applications.

You can view a log summary generated using the logparse utility. For more information, see *Siebel Database Upgrade Guide* .

Troubleshooting Siebel Repository Import for the Siebel Database

This task is part of *Troubleshooting Installation for the Siebel Database*.

Typical problems that can occur at this stage are shown in the following table. These errors might appear in the log files produced by the Siebel Repository import process.

Problem	Cause	Solution
Cannot import a Siebel Repository	Importing a Siebel Repository with the same name as an existing Siebel Repository	Choose a unique name for the new Siebel Repository.
Database runs out of tablespace pages and cannot allocate new data pages	Insufficient tablespace size	Increase the tablespace sizes.

Problem	Cause	Solution
(IBM DB2 only) Errors regarding the settings for DB2 configuration parameters, such as APP_CTL_ HEAP_SZ	Values too low	Reset the DB2 configuration parameters to higher values. Guidelines for Siebel CRM suggest using only minimum values.

Verifying Installation for Siebel Tools

After installing Siebel Tools in a new installation, update installation, or migration installation, review the directory structure created by the installer for the Siebel Tools, review the installation log files and review the Siebel Tools directories and files that now exist under the `SIEBEL_TOOLS_ROOT` directory that you specified during the installation. In addition to the steps described in this topic and elsewhere, perform any other tasks that might be necessary for verifying that your Siebel CRM software is functioning correctly.

This topic includes the following information:

- *Reviewing the Installation Log Files for Siebel Tools*
- *Reviewing the Installation Directories for Siebel Tools*
- *Verifying Successful Installation of Siebel Tools*
- *Verifying Directory Permissions for Siebel Tools*
- *Verifying the ODBC Data Sources for Siebel Tools*

Related Topics

About Installing Siebel Web Client or Siebel Tools

Reviewing the Installation Log Files for Siebel Tools

This task is part of *Verifying Installation for Siebel Tools*.

After installing Siebel Tools, review the installation log files to verify that all of the components installed successfully, particularly if you experienced any difficulties with the installation. The log files are created in the following directory:

```
SIEBEL_TOOLS_ROOT\cfgtoollogs\oui
```

The relevant log files for an installation session are named installActionsdate_time.log file. In the file name, date_time represents the date and time when the file was created; for example, installActions2021-01-12_10-28-04PM.log. For each set of files created, the oldest and largest log file generally represents the Siebel CRM module being installed (such as Siebel Tools).

Reviewing the Installation Directories for Siebel Tools

This task is part of *Verifying Installation for Siebel Tools*.

The following minimum directories are created for a Siebel Tools installation. These directories, the files and subdirectories they contain, and various other files are created in the directory that you specified during the installation, such as `C:\Siebel\Tools`. Use Windows Explorer to verify the directory structure on your computer. In the list, lang represents each language that you have installed, such as `ENU` for U.S. English.

In this guide, SIEBEL_TOOLS_ROOT represents the directory for the Siebel Tools installation, such as `C:\Siebel\Tools` (or a similar location). This installation directory contains subdirectories such as the ones identified here. You can change the directory name during the installation process. For more information, see *Requirements for Installing Siebel Tools*.

Note: If you install Siebel Tools with other modules, such as Siebel Enterprise Components or Siebel Web Client, then some of the directories present after installation are used by the colocated modules. For more information, see *Reviewing the Installation Directories for Siebel Enterprise Components*.

Directory	Description
BIN	Directory containing binary files (such as BAT, BIN, DLL, or EXE files), CFG files (located in language subdirectories), DSN files, SQL files, user preference files, language-specific files, and other files used by Siebel Tools. This directory might also be used by Siebel Web Client.
classes	Java code files: examples examples\src examples\src\com examples\src\com\siebel examples\src\com\siebel\extra examples\src\com\siebel\integration examples\src\com\siebel\integration\mq (Examples of Java code files.) examples\src\com\siebel\integration\servlet (Examples of Java code files.)
help	Siebel Tools help files: lang. Help files for the language that you installed.
locale	Text files for installed languages containing version information.
log	Log files from Siebel Tools operations.
msgtempl	Message files.
public	Contains the default HTML file (default.htm) and also subdirectories such as `files`, `fonts`, `htmltemplates`, `images`, and `scripts`. CSS files are located in the `files` directory, for example.
reppatch	Location of the XML files used by the maintenance update wizard. All log files generated by this wizard are also created under `reppatch\log`.

Directory	Description
scripts	Location of Java scripts.
sqltempl	SQL statement fragments used by certain Siebel Tools operations.
temp	Temporary working area.

Verifying Successful Installation of Siebel Tools

This task is part of *Verifying Installation for Siebel Tools*.

After installing Siebel Tools, verify that installation was successful by starting Siebel Tools and connecting to the Siebel database.

To verify that the Siebel Tools installation completed successfully

- Start Siebel Tools and log in to the Siebel database.

Verifying Directory Permissions for Siebel Tools

This task is part of *Verifying Installation for Siebel Tools*.

After installing Siebel Tools, verify that the Siebel Tools user has sufficient permissions to read and write data to the Siebel Tools installation directory.

To verify directory permissions for Siebel Tools

1. Select the Siebel Tools root directory and right-click on it.
2. Choose Properties and go to the Sharing tab.
3. Make sure that the Siebel Tools user has read and write access.

In addition, in user account properties, the Siebel Tools user must not be defined as a restricted user in the group membership section.

Verifying the ODBC Data Sources for Siebel Tools

This task is part of *Verifying Installation for Siebel Tools*.

The Siebel Tools installer creates the ODBC data sources described in the following table. By default, these are created as system data sources, which are visible to all of the user accounts on the computer where Siebel Tools is installed.

Data Source	Use
SSD default instance	Connects to all non-DB2 database platforms.
SSD	Connects to the IBM DB2 database.

For Oracle Database, verify that the following registry keys are set as shown. These values are required in order for the ODBC driver to behave correctly.

```
LobPrefetchSize=8388608
FetchBufferSize=1048576
```

For more information about how to run the Microsoft ODBC Administrator to verify ODBC data sources, see *Verifying the ODBC Data Source*. See also *Requirements for Installing Siebel Tools*.

Verifying Installation for the Siebel Web Client

After installing Siebel Web Client in a new installation, update installation, or migration installation, review the installation log files and review the Siebel Web Client directories and files that now exist under the `SIEBEL_CLIENT_ROOT` directory that you specified during the installation. In addition to the steps described in this topic and elsewhere, perform any other tasks that might be necessary for verifying that your Siebel CRM software is functioning correctly.

This topic includes the following information:

- *Reviewing the Installation Log Files for the Siebel Web Client*
- *Reviewing the Installation Directories for the Siebel Web Client*
- *Verifying ODBC Data Sources for the Siebel Developer Web Client*

Related Topics

About Installing Siebel Web Client or Siebel Tools

Reviewing the Installation Log Files for the Siebel Web Client

This task is part of *Verifying Installation for the Siebel Web Client*.

After installing the Siebel Web Client software, review installation log files to verify that all of the components installed successfully, particularly if you experienced any difficulties with the installation. The log files are created in the following directory:

```
SIEBEL_CLIENT_ROOT\cfgtoollogs\oui
```

The relevant log files for an installation session are named installActionsdate_time.log file. In the file name, date_time represents the date and time when the file was created; for example, installActions2021-01-12_10-28-04PM.log. For each set of files created, the oldest and largest log file generally represents the Siebel CRM module being installed, such as Siebel Web Client. This topic applies to the Siebel Web Client installation, optionally with the local database.

Reviewing the Installation Directories for the Siebel Web Client

This task is part of *Verifying Installation for the Siebel Web Client*.

The following minimum directories are created for a Siebel Web Client installation. These directories, the files and subdirectories they contain, and various other files are created in the directory that you specified during the installation, such as `C:\Siebel\Client`. Use Windows Explorer to verify the directory structure on your computer.

In this guide, SIEBEL_CLIENT_ROOT represents the directory for the Siebel Web Client installation, such as `C:\Siebel\Client` (or a similar location). This installation directory contains subdirectories such as the ones identified here. You can change the directory name during the installation process. For more information, see *Requirements for Installing Siebel Web Clients*.

Note: If you install Siebel Web Client with other modules, such as Siebel Enterprise Components or Siebel Tools, then some of the directories present after installation are used by the colocated modules. For more information, see *Reviewing the Installation Directories for Siebel Enterprise Components*.

Directory	Description
BIN	Directory containing binary files (such as BAT, BIN, DLL, or EXE files), CFG files (located in language subdirectories), DSN files, SQL files, user preference files, language-specific files, and other files used by Siebel Web Client. This directory might also be used by Siebel Tools.
fonts	Contains font files.
lex	Contains dictionary files used by the spelling checker.
local	Where the local database is stored.
locale	Stores language-specific files.
log	Stores the log files from Siebel Web Client operations, such as synchronization.
msgtempl	Stores message files used by the Siebel Web Client.
oraclese	The installation path for Oracle Database SE2. This directory is created when you installed that module with the Siebel Web Client, as described in *Installing and Using Oracle Database SE2 for the Local Database*.
public	Contains the default HTML file (default.htm) and also subdirectories such as **`files`**, **`fonts`**, **`htmltemplates`**, **`images`**, and **`scripts`**. CSS files are located in the **`files`** directory, for example.
sqltempl	Contains SQL scripts. Do not modify these files.
temp	Contains working report files.

Directory	Description
upgrade	Contains Siebel Anywhere upgrade files that have been retrieved by the user.

Verifying ODBC Data Sources for the Siebel Developer Web Client

This task is part of *Verifying Installation for the Siebel Web Client*.

Verification of ODBC data sources for the Siebel Developer Web Client is similar to that for Siebel Tools. For information, see *Verifying the ODBC Data Sources for Siebel Tools*.

Verifying Installation for Oracle Database SE2

Oracle Database SE2 is installed with Siebel Mobile Web Client. You can verify the existence of or the successful installation of Oracle Database SE2 in multiple ways. This topic includes the following information:

- *Verifying Installation of Oracle Database SE2*
- *Verifying the ODBC Data Source for Oracle Database SE2*
- *Verifying the TNSNAMES.ORA Entry for Oracle Database SE2*

Related Topics

Installing and Using Oracle Database SE2 for the Local Database

Verifying Installation of Oracle Database SE2

This task is part of *Verifying Installation for Oracle Database SE2*.

Use the following methods to verify the installation of Oracle Database SE2 for Siebel Mobile Web Client. Also perform the other verification steps in this topic. See also the documentation for Oracle Database Standard Edition 2 (Oracle Database SE2) on Oracle Help Center.

To verify installation of Oracle Database SE2

- Do any or all of the following:
  - Open SERVICES.MSC and check for OracleServiceSE.
  - Open Programs and Features and check for Oracle Database SE2 under installed programs.
  - Check for the presence of the `oraclese` directory in the Siebel Mobile Web Client installation.
  - Open the install.log file, which is available under the `%temp%` folder. Make sure that [ResultCode] is set to zero.
  - Run the following command to find the entry for Oracle Database SE2 in the registry:

    ```
    reg query HKLM\SOFTWARE\ORACLE\KEY_SE /v ORACLE_BASE
    ```

  - Start Siebel Mobile Web Client and connect to the LOCAL_SE data source.

Note: Connecting to LOCAL_SE for the first time initializes the local database. Users must attempt this only after a local database has been extracted for them by the system administrator.

Verifying the ODBC Data Source for Oracle Database SE2

This task is part of *Verifying Installation for Oracle Database SE2*.

Installing Oracle Database SE2 with Siebel Mobile Web Client automatically creates an ODBC system data source name (DSN) that Siebel Mobile Web Client uses to connect to an Oracle Database SE2 installation. The ODBC data source is LOCAL_SE for a Siebel Mobile Web Client installation, with the local database using Oracle Database SE2.

To verify the ODBC data source for Oracle Database SE2

1. Open ODBCAD32.exe, which is available in the `c:\Windows\SysWOW64` directory.
2. Check that one of the following ODBC data sources has been added under the System DSN tab, corresponding to your installation: LOCAL_SE.
3. Make sure that the DSN connects.

Verifying the TNSNAMES.ORA Entry for Oracle Database SE2

This task is part of *Verifying Installation for Oracle Database SE2*.

Use the following procedure to verify the TNSNAMES.ORA entry for Oracle Database SE2.

To verify the TNSNAMES.ORA entry for Oracle Database SE2

1. Open TNSNAMES.ORA, which is available under `Oracle_DB_Client\network\admin`.
2. Verify that a new entry of LOCAL_SE has been added.

Verifying Network Connectivity for the Siebel Server Computer

The Siebel Server must have network access to other Siebel CRM modules, such as the Siebel Gateway, the Siebel Application Interface, the Siebel database on the RDBMS, and the Siebel File System. Use the following procedure to verify that the Siebel Server computer can access other Siebel CRM modules.

> **Note:** For information about verifying network connectivity for IBM DB2 for z/OS, see *Implementing Siebel Business Applications on DB2 for z/OS* .

To verify network connectivity for the Siebel Server computer

1. Verify network connectivity to the Siebel Gateway, the Siebel Application Interface, and the Siebel database from the Siebel Server computer, using the test utility for your network type.

   For TCP/IP networks, use the `ping` utility to verify network connectivity. Or test the connection through Control Panel, Administrative Tools, Data Sources (ODBC), and then System DSN.
2. For Oracle Database, verify connectivity to the Siebel database, then go to Step 5.

   Use the `tnsping` utility and Oracle Database connectivity alias, as appropriate to your database, from a command prompt window to make sure that you can connect to the database using the network connect string that you defined.
3. For IBM DB2, verify connectivity to the Siebel database, as follows:

   a. Open a DB2 Command Window to make sure that you can connect to your database.

      > **CAUTION:** Use the DB2 Command Window, not the DB2 command line processor, to enter these commands, because the DB2 command line processor window uses different syntax. The commands in this procedure work only if you issue them in a DB2 Command Window.

   b. Choose Start, All Programs, IBM DB2, Command Line Tools, and then Command Window. Enter:

      ```
      db2 connect to database_alias user user_ID using password
      ```

      where:

      - database_alias is your database alias
      - user_ID is a valid user name on DB2
      - password is the password for that user_ID

      If your connection is valid, then you see a message that looks like the following:

      ```
      Database Connection Information

      Database Server = DB2/NT x.x.x
      SQL authorization ID = SADMIN
      Database alias = DB_Alias
      ```

c. If your connection is not valid, then verify your configuration.
d. To close the connection, type `db2 terminate`. You can also use the DB2 Command Center GUI tool to terminate the connection.

4. For Microsoft SQL Server/Azure SQL Database Server, verify connectivity to the Siebel database, then go to the next step.

   Navigate to Control Panel, Administrative Tools, and then Data Sources (ODBC). Under System DSN, select the data source, and click Configure. This wizard provides a button for testing the connection.
5. Provide network connections from the Siebel Server to all of the applicable Siebel File System directories.

   Do not use a mapped network drive as this network connection. Instead, use a UNC share. The sharename syntax must be in this form: `\\Siebel_Server_Name\Sharename` .

   For more information, see *Creating the Siebel File System*.
6. Verify that the Siebel File System directories are visible and that the Siebel service owner account has the necessary permissions on these directories.

   If you are clustering the Siebel File System, then use the Network IP resource or Network Hostname Resource assigned to that cluster file share.

Verifying Network Connectivity for Mobile Users

Siebel Mobile Web Client users must be able to connect to the Siebel Remote Server, using TCP/IP, to synchronize with the master Siebel database. Make sure that you have the correct network software and hardware installed to support this connectivity and that your remote users are able to establish a TCP/IP connection to the server. To check connectivity, use the `ping` utility.

Related Books

Siebel Remote and Replication Manager Administration Guide

Configuring the Siebel Mobile Web Client When Neither Siebel VB nor Siebel eScript Is Licensed

By default, the Siebel Mobile Web Client has Siebel VB or Siebel eScript enabled. The parameter EnableScripting is set to True in the application configuration files. If this parameter is True, but neither Siebel VB nor Siebel eScript is licensed, then the Siebel Mobile Web Client does not start. An error message is returned, indicating that you must turn off Siebel VB or Siebel eScript before running the Siebel Mobile Web Client.

To configure the Siebel Mobile Web Client when neither Siebel VB nor Siebel eScript is licensed

- Set EnableScripting to FALSE in the configuration file used by your Siebel Mobile Web Client, such as uagent.cfg for Siebel Call Center. This configuration file is located in the directory `SIEBEL_CLIENT_ROOT\bin\LANGUAGE` .

where:

- `SIEBEL_CLIENT_ROOT` is the directory where you installed the Siebel Mobile Web Client.
- LANGUAGE is the three-letter code for the language in which you are running the Siebel Mobile Web Client.

9 Uninstalling Siebel CRM

Uninstalling Siebel CRM

This chapter provides instructions for uninstalling Siebel CRM. It includes the following topics:

- *About Uninstalling Siebel CRM*
- *Process of Removing Configuration Data*
- *Uninstalling Siebel CRM*
- *Uninstalling Oracle Database XE*
- *About Uninstalling Earlier Versions of Siebel CRM*

About Uninstalling Siebel CRM

The topics in this chapter describe how to uninstall the current release of Siebel CRM. This chapter describes how to uninstall the software as well as how to restore a prior release.

This topic includes the following information:

- *Options for Uninstalling Siebel CRM*
- *Requirements for Uninstalling Siebel CRM*
- *Limitations for Uninstalling Siebel CRM*

Related Topics

Installation-Related Changes in Siebel CRM

Uninstalling Siebel CRM

About Uninstalling Earlier Versions of Siebel CRM

Related Books

Siebel Deployment Planning Guide

Siebel System Administration Guide

Siebel Global Deployment Guide

Options for Uninstalling Siebel CRM

This topic is part of *About Uninstalling Siebel CRM*.

This topic summarizes the uninstallation options for the Siebel CRM modules described in this guide. The following uninstallation options apply for Siebel CRM modules. For details, see *Uninstalling Siebel CRM*.

- **Full uninstallation.** Removes the installed Siebel CRM software module. For details, see *Performing a Full Uninstallion*.
- **Restoring a prior release.** Restores an installation of a prior release that uses the new installer (Siebel CRM 21.2 or later). For details, see *Restoring a Prior Installation and Configuration*.

  > **Note:** You can also restore a prior installation that does not use the new installer (Siebel CRM 21.1 or earlier). The release that you can restore (or roll back to) and the specific steps you must take depend on how you had originally installed the Siebel CRM software modules. Different methods are provided for different situations. Make sure to use the correct method for your situation. For more information about this task, refer to the rollback uninstallation instructions in the *Siebel Installation Guide* for your operating system for your prior release, such as Siebel CRM 21.1 or earlier.

Related Topics

Requirements for Uninstalling Siebel CRM

Limitations for Uninstalling Siebel CRM

Uninstalling Siebel CRM

Requirements for Uninstalling Siebel CRM

This topic is part of *About Uninstalling Siebel CRM*.

Uninstallation of Siebel CRM software is subject to various requirements. The following requirements apply to uninstalling Siebel CRM software:

- **Use documented methods.** To successfully uninstall, you must use the supported methods documented in this chapter and observe all of the requirements and limitations that apply for uninstalling each module. Review this entire chapter before you uninstall any software.
- **Before a full uninstallation, you must remove configuration data.** For Siebel CRM server modules, including Siebel Server, Siebel Gateway, Siebel Application Interface, or other modules that were previously configured using Siebel Management Console, you must perform tasks to remove the configuration data before you uninstall the Siebel software. For detailed information, see *Process of Removing Configuration Data*.
- **You must have the required access.** When you uninstall an instance of a Siebel CRM server module, you must either be the same user or be a user in the same group as the user who performed the installation. This requirement applies to both full uninstallation and restoring a prior installation.
- **Clustered nodes present special requirements.** Clustered nodes present special requirements that might change part of the installation and uninstallation process from what is described in this guide. Siebel Gateway clustering includes such requirements, as noted in *Changing the Primary Nodes for a Siebel Gateway Cluster*.
- **Do not delete installation directories until you have performed a full uninstallation.** Do not delete the installation directories for any Siebel CRM module until after you have performed a full uninstallation of the module. After a full uninstallation has completed successfully, you might have to restart your computer. Afterwards, you can safely delete the remaining directories. (Do not delete any installation directories after you perform an uninstallation as part of restoring a prior installation.)
- **Siebel CRM modules must use the same version after restoring an installation of a prior version.** If you restore a previously installed version of Siebel Enterprise Server and SWSE, then you must also perform a

similar process for all of the other Siebel CRM modules sharing the same Siebel Enterprise and Siebel database. After restoring the prior installation, see the documentation that applies to your currently installed Siebel CRM software version for information about this version.

- **To restore a prior release, you must have retained the previous installation.** If you installed the current release as a migration installation from Siebel CRM 16.x or earlier, then the existing installation directory was renamed. You must retain this installation directory in order to be able to restore the version that it represents. For information about restoring or rolling back to earlier releases, see the applicable *Siebel Installation Guide* for your operating system for the release you are restoring.
- **After restoring a prior release, you must restore the Siebel Repository.** New functionality and bug fixes might no longer be available after restoring a prior release. After you restore a prior release, you must remove any configuration changes and restore the Siebel Repository and the Siebel Repository file (SRF), where applicable, to the state when you first installed the current release. (SRF files are applicable only to releases prior to Siebel CRM 17.0.)

  > **Note:** After you installed the current release as a migration installation, you would have used the Incremental Repository Merge feature to update the Siebel Repository and to make other Siebel database changes. For more information about Incremental Repository Merge, see *Siebel Database Upgrade Guide* .

- **After restoring a prior release, you must re-enable the applicable services.** After you restore a prior release, the Siebel Server service and the Siebel Gateway Name Server service are disabled and must be re-enabled.
- **After restoring a prior release, refer to earlier versions of the documentation.** After you restore a prior release, you must refer to the documentation that applies to the Siebel CRM version that you restore. You can find the *Siebel Installation Guide* for your operating system for prior releases on the *Siebel Bookshelf* for each release.
- **After restoring a prior release, verify the software version.** After restoring a prior release, verify the version number of the installed software that you restored. See the documentation that applies to your new effective version. See also *Verifying Installation for Siebel CRM* and related topics.
- **After restoring a prior release, you must restore previous required versions of third-party software.** After restoring a version prior to Siebel CRM 17.0, you must restore installed third-party (or non-Siebel Oracle) software to applicable prior versions supported for the version of Siebel CRM that you are restoring, where necessary.
- **After restoring a prior release, you must reconfigure SWSE.** After restoring a version prior to Siebel CRM 17.0, you must reconfigure the SWSE. For more information, see *Additional Tasks for Migration Installations* and *About Configuring Siebel CRM*.
- **After restoring a prior release, configure Oracle Configuration Manager (if necessary).** If Oracle Configuration Manager was not configured before you performed a migration installation of Siebel Enterprise Server, then, after restoring the original installation, you must configure this instance of Oracle Configuration Manager manually in order for it to function properly.

The following recommendations apply to performing a full uninstallation:

- **Uninstall Database Configuration Utilities only after backing up files.** Before you uninstall an instance of Siebel Enterprise Server that includes the Database Configuration Utilities and Siebel Server, it is strongly recommended that you back up relevant files, such as summary.html and summary.txt in the `SIEBEL_ROOT\dbsrvr` directory and the files in the `SIEBEL_ROOT\siebsrvr\log` directory.
- **Uninstall Database Configuration Utilities only after removing configuration data.** Before you uninstall the Database Configuration Utilities and the Siebel Server it is installed with, it is strongly recommended that you first remove the Siebel Enterprise configuration or remove both the Enterprise and the Siebel Gateway.
- **Uninstall Oracle Database SE2 before you perform a full uninstallation for Siebel Web Client.** If Oracle Database SE2 was installed for use with Siebel Web Client, and you plan to uninstall Siebel Web Client, then it

is recommended that you first uninstall Oracle Database SE2. For more information, see *Uninstalling Oracle Database XE*.

Related Topics

Installing and Deploying Siebel CRM with Multiple Languages

Changing the Primary Nodes for a Siebel Gateway Cluster

Options for Uninstalling Siebel CRM

Limitations for Uninstalling Siebel CRM

Uninstalling Siebel CRM

Related Books

Siebel Database Upgrade Guide

Limitations for Uninstalling Siebel CRM

This topic is part of *About Uninstalling Siebel CRM*.

Uninstallation of Siebel CRM software is subject to various limitations. The following limitations apply to uninstalling Siebel CRM software:

- **You cannot selectively uninstall Siebel Enterprise Server modules installed together on the same computer.** Although all Siebel CRM server modules (including Siebel Gateway, Siebel Server, and Siebel Application Interface) are always installed together, you might have only configured certain modules on certain installations, or you might only be using certain modules. Even if you are not using certain modules in a particular installation, you cannot uninstall just those modules. If you do not want to use a particular Siebel Server, then you can remove configuration data for this Siebel Server or disable the Siebel Server so that it does not run on this server computer.

  For information about disabling Siebel Server components, see *Siebel System Administration Guide* . See also *Preparing to Run Siebel Server Components*.

- **You cannot selectively uninstall Siebel languages.** If you previously installed a particular Siebel language but determine that it is not needed on that installation, then you can deploy your applications using only the languages that you require. You cannot uninstall the Siebel language that you do not need.

  If you have already deployed languages that you do not require, then you can remove and recreate a Siebel Server configuration with fewer deployed languages, or recreate a Siebel Application Interface profile with fewer deployed languages and reapply it to each installed Siebel Application Interface instance. Alternatively, you can disable unneeded language-specific Application Object Managers or other components. For these tasks, you do not uninstall the Siebel software. For more information about multilingual deployments, see *Installing and Deploying Siebel CRM with Multiple Languages* and *Siebel Global Deployment Guide* .

- **Limitations might apply for previous versions that you restore or roll back to.** If you installed the current release as a migration installation, you can restore the previously installed release. For more information, see *About Uninstalling Earlier Versions of Siebel CRM*.

- **This guide describes uninstalling the current release only.** For information about uninstalling any prior release of Siebel CRM, see *About Uninstalling Earlier Versions of Siebel CRM*.

- **This guide describes restoring a prior release that uses the new installer.** For restoring or rolling back to earlier releases, see the applicable *Siebel Installation Guide* for your operating system for the release you are restoring.
- **This guide describes GUI uninstallation only.** Topics in this chapter describe full uninstallation by running Oracle Universal Installer in GUI mode. It is also possible to perform a full uninstallation by using silent mode, similar to the installation process described in *Installing Siebel CRM Using Silent Mode*.

Related Topics

Installing and Deploying Siebel CRM with Multiple Languages

Options for Uninstalling Siebel CRM

Requirements for Uninstalling Siebel CRM

About Uninstalling Earlier Versions of Siebel CRM

Related Books

Siebel System Administration Guide

Siebel Global Deployment Guide

Siebel CRM Update Guide and Release Notes on My Oracle Support

Process of Removing Configuration Data

This topic describes the tasks that you perform for removing the configuration data for Siebel CRM server modules, such as Siebel Server or Siebel Application Interface. You perform these tasks using the Siebel Management Console, which you used for configuration.

Removing configuration data is related to uninstalling the Siebel CRM software, but it is distinct from it. Before you uninstall Siebel CRM server modules, including Siebel Server, Siebel Application Interface, or Siebel Gateway, you must remove configuration data that was previously created and deployed for these modules. In some cases, you might choose to remove configuration data without uninstalling. Removing configuration data does not affect the profiles that you used when deploying the installed modules.

This topic does not describe removing configuration profiles that you have created but have not deployed. For more information, see *Running the Siebel Management Console*.

Perform the tasks for removing configuration data in the following order (the reverse of the normal order of configuration):

1. *Removing the Siebel Application Interface Configuration*
2. *Removing the Siebel Server Configuration*
3. *Removing the Siebel Enterprise*
4. *Removing the Siebel Gateway Cluster*
5. *Removing the Siebel Gateway*

Related Topics

Configuring Siebel CRM Server Modules

Running the Siebel Management Console

Additional Tasks for Migration Installations

About Uninstalling Siebel CRM

Uninstalling Siebel CRM

Removing the Siebel Application Interface Configuration

This task is a step in *Process of Removing Configuration Data*.

The Siebel Management Console lets you remove the configuration that you deployed for the Siebel Application Interface. This task removes the application configuration data that was created when you deployed the Siebel Application Interface profile.

Note: This task applies only to instances of Siebel Application Interface that are in a deployed state.

To remove the Siebel Application Interface configuration

1. Run the Siebel Management Console, as described in *Running the Siebel Management Console*.
2. Click Siebel Deployment in the navigation menu in the side panel.
3. In the hierarchy shown, click Application Interface, under Siebel Deployment.
4. Click to select the box for the deployed instance of Siebel Application Interface for which you want to remove the configuration. Deployed instances show a solid green circle icon (whose tooltip text reads Deployed).
5. Click the Delete icon, whose tooltip text reads Delete Application Interface.

   If the delete step is successful, then the deployment item disappears. If not, then an error message is displayed.

Removing the Siebel Server Configuration

This task is a step in *Process of Removing Configuration Data*.

The Siebel Management Console lets you remove Siebel Server configuration data from the Siebel Gateway. This task removes the Siebel Server service from the local server computer.

Note: When removing Siebel Server configuration data using this task, observe all of the documented requirements. Perform this task only if you have uninstalled, or removed all of the applicable configuration data for, all of the installed instances of Siebel Application Interface that depend on this Siebel Server. Before you uninstall a previously configured instance of Siebel Server, you must perform this task. This task can also be useful in cases that do not involve uninstallation. When you perform this task, the Siebel Gateway must be running.

If this Siebel Server was not configured, then this task is not necessary before uninstallation.

To remove the Siebel Server configuration

1. Make sure that the Siebel Gateway is running.
2. Run the Siebel Management Console, as described in *Running the Siebel Management Console*.
3. Click Siebel Deployment in the navigation menu in the side panel.
4. In the hierarchy shown, click the Siebel Server box, under Siebel Deployment.

5. Click to select the box for the deployed instance of Siebel Server for which you want to remove the configuration. Deployed instances show a solid green circle icon (whose tooltip text reads Deployed).
6. Click the Delete icon, whose tooltip text reads Delete Siebel Server.

   If the delete step is successful, then the deployment item disappears. If not, then an error message is displayed.

Removing the Siebel Enterprise

This task is a step in *Process of Removing Configuration Data*.

The Siebel Management Console lets you remove the Siebel Enterprise for a Siebel deployment.

> **Note:** When removing a Siebel Enterprise using this task, observe all of the documented requirements. Perform this task only if you have uninstalled, or removed all of the applicable configuration data for, all of the installed instances of Siebel Application Interface and Siebel Server that depend on this Siebel Enterprise. When you perform this task, the Siebel Gateway must be running.

To remove the Siebel Enterprise

1. Make sure that the Siebel Gateway is running.
2. Run the Siebel Management Console, as described in *Running the Siebel Management Console*.
3. Click Siebel Deployment in the navigation menu in the side panel.
4. In the hierarchy shown, click the Enterprise box, under Siebel Deployment.
5. Click to select the box for the deployed Siebel Enterprise that you want to remove. Deployed instances show a solid green circle icon (whose tooltip text reads Deployed).
6. Click the Delete icon, whose tooltip text reads Delete Enterprise.

   If the delete step is successful, then the deployment item disappears. If not, then an error message is displayed.

Removing the Siebel Gateway Cluster

This task is a step in *Process of Removing Configuration Data*.

The Siebel Management Console lets you remove the Siebel Gateway cluster for a Siebel deployment.

> **Note:** When removing a Siebel Gateway cluster using this task, observe all of the documented requirements. When you perform this task, the primary Siebel Gateway must be running.

Related Topics

Configuring the Siebel Gateway Cluster

Removing the Siebel Gateway

To remove the Siebel Gateway cluster

1. Make sure that the primary Siebel Gateway is running.
2. Run the Siebel Management Console, as described in *Running the Siebel Management Console*.

3. Click Siebel Deployment in the navigation menu in the side panel.
4. In the hierarchy shown, click the Gateway Cluster box, under Siebel Deployment.
5. Click to select the box for the deployed Siebel Gateway cluster. Deployed instances show a solid green circle icon (whose tooltip text reads Deployed).
6. Click the Delete icon, whose tooltip text reads Delete Gateway Cluster.

   If the delete step is successful, then the deployment item disappears. If not, then an error message is displayed.

Removing the Siebel Gateway

This task is a step in *Process of Removing Configuration Data*.

The Siebel Management Console lets you remove the Siebel Gateway service from the server computer, for a Siebel Gateway that you have installed and configured.

> **Note:** When removing a Siebel Gateway using this task, observe all of the documented requirements. Perform this task only if you have uninstalled, or removed all of the applicable configuration data for, all of the installed instances of Siebel Application Interface and Siebel Server that depend on this Siebel Gateway. Also remove the Siebel Gateway cluster, if configured. Before you uninstall a previously configured instance of Siebel Gateway, you must perform this task. When you perform this task, the Siebel Gateway must be running.

If this Siebel Gateway was not configured, then this task is not necessary before uninstallation.

To remove the Siebel Gateway

1. Make sure that the Siebel Gateway is running.
2. Run the Siebel Management Console, as described in *Running the Siebel Management Console*.
3. Click Settings in the navigation menu in the side panel.

   The Siebel Management Console displays the URL for the physical instance of Siebel Gateway for the Siebel deployment.
4. Check the box labeled *I understand that removing Gateway will remove all profiles and deployments registered with the Gateway and remove the Gateway services.*
5. Click Remove Gateway.

Uninstalling Siebel CRM

This topic describes how to uninstall Siebel CRM for the current release. Both full uninstallation and restoring an installation of a prior release are described.

In a full uninstallation, the installed Siebel Enterprise components are uninstalled. The following modules are uninstalled where present in this installation: Siebel Enterprise Components (Siebel Application Interface, Siebel Gateway, Siebel Server, Database Configuration Utilities, EAI Connector), Siebel Web Client, Siebel Tools, and other modules. (Some of these modules might have been explicitly installed first for a release prior to Siebel CRM 21.2. Individual server modules are no longer selectable in the new installer.) Some modules might have been installed on a single computer or operating system instance, or different modules might have been installed on multiple computers or operating system instances. Separate uninstallation tasks are required for each installed instance.

CAUTION: Before you proceed with any uninstallation task, review all of the information in *About Uninstalling Siebel CRM*, particularly the requirements and limitations.

This topic includes the following information:

- *Performing a Full Uninstallion*
- *Restoring a Prior Installation and Configuration*

Performing a Full Uninstallation

This task is part of *Uninstalling Siebel CRM*.

You can perform a full uninstallation of Siebel CRM, whether you installed it as a new installation or as a migration or update from a prior release. To uninstall this installation of Siebel Enterprise Components or Siebel Clients and remove all of the Apache Tomcat services, run `SIEBEL_ROOT\uninstaller.bat` (Windows) or `SIEBEL_ROOT/uninstaller` (UNIX). Other requirements might apply.

- (Windows) SIEBEL_ROOT\oui\bin\setup.exe
- (UNIX) SIEBEL_ROOT/oui/bin/runInstaller.sh -invPtrLoc inventory_file_location

Uninstall the current release of Siebel CRM using the Siebel CRM uninstaller function of the Siebel CRM installer, as follows for your operating system:

Note: Before a full uninstallation, you must remove configuration data. For Siebel CRM server modules (including Siebel Application Interface, Siebel Server, Siebel Gateway) that were previously configured using Siebel Management Console, you must perform tasks to remove the applicable configuration data before you uninstall the Siebel software.

If, instead, your intention is to restore an installation of a prior release of Siebel CRM that supports the new installer and restore your prior configuration, then you can optionally retain the configuration data before uninstalling if this data will be valid for the prior release as well. You must at least stop the services for the Siebel Gateway and Siebel Server. For more information, see *Restoring a Prior Installation and Configuration*.

To perform a full uninstallation

1. Review all of the information in *About Uninstalling Siebel CRM*, particularly the requirements and limitations.
2. Verify that you have performed any required tasks to uninstall other modules or remove configuration data.

   Note: For example, before you uninstall an installed instance of Siebel CRM server software, whether it includes Siebel Gateway, Siebel Server, Siebel Application Interface, or other components, you must perform all of the applicable Siebel Management Console tasks to remove existing configuration data. For more information, see *Process of Removing Configuration Data*.

3. Choose Start, Products, Oracle - SES_HOME (or a similar Oracle home identifier), Oracle Installation Products, and then Universal Installer.

   This action starts the instance of Oracle Universal Installer that was installed with your Siebel Enterprise Server installation.
4. Click Installed Products.
5. In the Inventory dialog box, select the Oracle Home value (for example, SES_HOME) representing the installed instance of Siebel CRM software that you want to uninstall.
6. Click Remove.

7. Click Yes to confirm that you want to uninstall this Siebel CRM software and all of its dependent modules on the same computer.
   The Siebel CRM software is uninstalled from this computer.
8. In the Inventory dialog box, click Close.
9. In the Oracle Universal Installer window, click Cancel.
10. Restart your server.

Related Topics

Siebel CRM Download and Installation

About Uninstalling Siebel CRM

Process of Removing Configuration Data

Restoring a Prior Installation and Configuration

This task is part of *Uninstalling Siebel CRM*.

Where you installed Siebel CRM as an update from a prior release, you can uninstall the software on a single computer or operating system instance and restore your prior release's installation and configuration.

Do one of the following, according to your requirements:

- Restore a prior installation (Siebel CRM 21.2 or later) that uses the new installer.
- Restore a prior installation (Siebel CRM 21.1 or earlier) that does not use the new installer.

Before you install Siebel CRM, it is strongly recommended to do the following:

- **Back up the Siebel database.** Back up the Siebel database so that, after installing Siebel CRM as described in *Installing Siebel CRM in an Update Installation*, allowing the PostInstallDBSetup utility to run, and optionally running RepositoryUpdate, if you decide to roll back the installation and restore your prior release, as described in this topic, then you can use this backup to restore the Siebel database to its prior state.
- **Back up the Siebel Gateway registry (ZooKeeper).** Back up the Siebel Gateway registry so that, after installing Siebel CRM as described in *Installing Siebel CRM in an Update Installation*, if you decide to roll back the installation and restore your prior release, as described in this topic, then you can use this backup to restore the Siebel Gateway registry to its prior state. Where possible, you can restore the Siebel Gateway registry from backup files created by the installer. Otherwise, restore from backups you made manually. For more information about backing up and restoring the Siebel Gateway registry, see *Siebel System Administration Guide* .

  > **Note:** As noted in *Siebel System Administration Guide* , an updated version of Apache ZooKeeper is provided for the Siebel Gateway registry. This new version introduced changes in the files and directories used by ZooKeeper. You must take these changes into account when restoring the Siebel Gateway registry as part of restoring your prior release.

Restoring a Prior Release That Uses the New Installer (Siebel CRM 21.2 or Later)

> **Note:** This uninstallation case applies where you have installed the current release of Siebel CRM as an update for Siebel CRM 21.2 and later decide to restore Siebel CRM 21.2.

You can restore a prior Siebel CRM release that uses the new installer (Siebel CRM 21.2 or later). When you perform an update installation from a prior release that supports the new installer, custom files in the installation directory are backed up, along with configuration files like server.xml and properties files, because the installer replaces these files due to product changes. You specify the backup directory location when you install the new release. By default, the backup directory location is `SIEBEL_ROOT\mde_backup`.

Within the backup directory, the version-specific backup subdirectory might be named `SIEBEL_ROOT_22.x`, such as `SIEBEL_CRM_22.x` (for an installation directory of `SIEBEL_CRM`). This directory contains only configuration files that might be used when restoring the prior release of Siebel CRM that supports the new installer.

The backup directory contains version-specific subdirectories containing the files that you might need to restore when you restore a prior version. You can restore required files only or all files. You must determine which files you require after the restore, based on whether the customizations and configurations are backward compatible.

To restore a prior release that uses the new installer (Siebel CRM 21.2 or later)

1. Verify that the current installation and installation history are qualified for performing this task.
2. Manually stop all of the services, as follows:
   - (Windows) Stop the services for Siebel Gateway, Siebel Server, and Apache Tomcat (for external and internal application containers).
   - (UNIX) Stop the services for Siebel Gateway and Siebel Server and the processes for Apache Tomcat (for external and internal application containers).

   Note: Do not remove any configurations previously created using Siebel Management Console.
3. Uninstall the current release of Siebel CRM using the OUI (Oracle Universal Installer) uninstaller function of the Siebel CRM installer, as follows for your operating system:
   - (Windows) `SIEBEL_ROOT\oui\bin\setup.exe`
   - (UNIX) `SIEBEL_ROOT/oui/bin/runInstaller.sh -invPtrLoc inventory_file_location`

   Note: Do not delete the installation directory or the files it contains, which include the backup directory.
4. Verify that the external and internal services exist after uninstallation and they are still in a stopped state.
5. Reinstall the prior release of Siebel CRM 21.2 or later to the same installation directory.

   Note: Do only the installation task; do not select configuration tasks.
6. Restore the Siebel database from the backup you made before performing the update installation.
7. Restore the Siebel Gateway registry from the backup you made before performing the update installation.
8. Manually start all of the services, as follows:
   - (Windows) Start the services for Apache Tomcat (for external and internal application containers), Siebel Gateway, and Siebel Server.
   - (UNIX) Start the processes for Apache Tomcat (for external and internal application containers) and the services for Siebel Gateway and Siebel Server.
9. Validate that the Siebel CRM application URLs work as expected and verify the application version.
10. Validate that your application customizations are retained. These customizations were preserved in the backup directory described at the start of this topic. You can manually apply the customizations that you require.

Restoring a Prior Release That Does Not Use the New Installer (Siebel CRM 21.1 or Earlier)

You can restore a prior Siebel CRM installation that does not use the new installer (Siebel CRM 21.1 or earlier). When you updated this prior installation to the current release of Siebel CRM, the backup directory was created with a name like `SIEBEL_ROOT_pre22.x.0.0.0`, such as `SES_pre22.x.0.0.0`, for example. This directory, which contains the complete installation, was created at the same level as the existing `SIEBEL_ROOT` directory. Follow the detailed steps for how to roll back to your prior release.

> **Note:** For a prior installation of Siebel Web Client (from Siebel CRM 20.8 through Siebel CRM 21.1), the backup that was created might be fairly large, because it also includes a full copy of Oracle Database SE2.

To restore a prior release that does not use the new installer (Siebel CRM 21.1 or earlier)

1. Stop the services for the current release.
2. Remove the home directories for the current release.
3. Rename the backup directories to actual home folders. (Backup directories were created as a complete backup when you installed the current release of Siebel CRM to update a prior release.)
4. As needed, recreate the services for prior versions of Tomcat.
5. Restore the Siebel database from the backup you made before performing the update installation.
6. Restore the Siebel Gateway registry from the backup you made before performing the update installation.
7. Start the services for the prior release.

Related Topics

Siebel CRM Download and Installation

Verifying Installation for Siebel CRM

About Uninstalling Siebel CRM

Process of Removing Configuration Data

Related Books

Siebel System Administration Guide

Uninstalling Oracle Database XE

Oracle Database XE was used for the local or sample database with previous versions of Siebel CRM. It is still installed with the Siebel Web Client installer in Siebel CRM 17.0, and is used through Siebel CRM 20.7. The local database now uses Oracle Database SE2 and the sample database is no longer supported. This topic describes how to uninstall Oracle Database XE when it is no longer needed.

> **Note:** Performing a full uninstallation of Siebel Mobile Web Client or restoring a prior release does not uninstall Oracle Database SE2. If you need to uninstall Oracle Database SE2, see the documentation for this product on Oracle Help Center.

Use the following procedure to uninstall Oracle Database XE. You can use this procedure if you previously installed Oracle Database XE and the local database or sample database with Siebel Web Client. Uninstalling Oracle Database XE also uninstalls the local database or the sample database. (You can also uninstall Oracle Database XE if you installed it for an earlier version of Siebel Tools.)

It is recommended that you uninstall Oracle Database XE if you installed it for use with Siebel Mobile Web Client but then determine that you will not use it. For example, you would uninstall Oracle Database XE if you uninstall Siebel CRM 17.0 or later and roll back to Siebel CRM 15.0. Uninstall Oracle Database XE before you uninstall or roll back Siebel Web Client. For full uninstallation of the current release or restoring a prior release, see *Uninstalling Siebel CRM*.

> **Note:** Performing a full uninstallation of Siebel Mobile Web Client or restoring a prior release does not uninstall Oracle Database XE. The local database or the sample database are not uninstalled. In addition, the ODBC data source created for Oracle Database XE is not deleted in these uninstallation cases.

As an alternative to the following procedure, you can use one of the following methods:

- Uninstall Oracle Database Express Edition from the Windows Control Panel.
- Uninstall Oracle Database Express Edition from Programs and Features.

After uninstalling Oracle Database XE, restart the computer.

To uninstall Oracle Database XE and the local database or sample database

1. In Windows Explorer, navigate to the Siebel image location for the Siebel release from which you installed Oracle Database XE with Siebel Web Client (Siebel CRM 20.7 and earlier). Then navigate to the directory where the Oracle Database XE installer is located.

   Navigate to the following location, based on the Siebel software for which you are uninstalling Oracle Database XE:

   - `Siebel_Image\Windows\Client\Siebel_Web_Client\Disk1\stage\XE\64\Disk1`

     In this path, `Siebel_Image` is the directory for your version-specific Siebel network image, such as `c:\Siebel_Install_Image\17.0.0.0`, for Siebel CRM 17.0.
2. To start the Oracle Database XE installer, double-click setup.exe.
3. Click Remove.
4. After uninstalling Oracle Database XE, restart the computer.
5. Delete the directory `SIEBEL_CLIENT_ROOT\oraclexe`.

Related Topics

Installing and Using Oracle Database SE2 for the Local Database

Uninstalling Siebel CRM

About Uninstalling Earlier Versions of Siebel CRM

To uninstall an earlier version of Siebel CRM software, you must use the documented uninstallation method for that version. For each version, always use the latest revision of the documentation that applies for the Siebel CRM version that you're uninstalling.

For Siebel CRM 21.1 and earlier, consult this guide for the operating system you're using.

For Siebel CRM 17.x through Siebel CRM 21.1

- For the full uninstallation or rollback uninstallation instructions for Siebel CRM 17.x (Siebel CRM 2017) and later, through Siebel CRM 21.1, see that version of this guide for the operating system you're using, which is the guide that you used when you installed Siebel CRM 17.0.

  If you're restoring a prior release (Siebel CRM 21.1 or earlier) that doesn't use the new Siebel CRM installer, see the rollback instructions in this guide. Look for your operating system in a recent release or for your prior release, such as Siebel CRM 21.1 or earlier.

For Siebel CRM 16.x

- For the full uninstallation or rollback uninstallation instructions for Siebel CRM 16.x (Siebel CRM 2016), see that version of this guide for the operating system you're using, which is the guide that you used when you installed Siebel CRM 16.0.

For Siebel CRM 15.x

- For the full uninstallation or rollback uninstallation instructions for Siebel CRM 15.x (Siebel CRM 2015), see that version of this guide for the operating system you're using, which is the guide that you used when you installed Siebel CRM 15.0.

For Siebel CRM 8.1.1.14 or Siebel CRM 8.1.1.11

- For the full uninstallation or rollback uninstallation instructions for Siebel CRM 8.1.1.14 (Siebel CRM 2014) or Siebel CRM 8.1.1.11 (Siebel CRM 2013), see the version of this guide for the operating system you're using, which is the guide that you used when you installed Siebel CRM 8.1.1.14 or 8.1.1.11.

For Siebel CRM 8.1.1 and Siebel CRM 8.1.1.x (through version 8.1.1.10)

- For the full uninstallation instructions for Siebel CRM 8.1.1, which used the InstallShield versions of the Siebel Enterprise Server and Siebel Web Server Extension (SWSE) installers, see the version of this guide for the operating system you're using, which is the guide that you used when you initially installed.
- For the full uninstallation instructions for Siebel CRM 8.1.1.8 through Siebel CRM 8.1.1.10 (when Siebel Enterprise Server and SWSE were installed as a new installation), see the version of this guide for the operating system you're using, which is the guide that you used when you initially installed.

For Siebel CRM 8.2.2.14 and Siebel CRM 8.2.2.4

- For the full uninstallation or rollback uninstallation instructions for Siebel CRM 8.2.2.14 (Siebel CRM 2014) or Siebel CRM 8.2.2.4 (Siebel CRM 2013), see the version of this guide for the operating system you're using, which is the guide that you used when you installed Siebel CRM 8.2.2.14 or 8.2.2.4.

For Siebel CRM 8.2.2 and Siebel CRM 8.2.2.x (through version 8.2.2.3)

- For the full uninstallation instructions for Siebel CRM 8.2.2, see the version of this guide for the operating system you're using, which is the guide that you used when you initially installed.

See *Oracle Siebel CRM Documentation* for older versions of the guides.

Related Topics

Installing and Deploying Siebel CRM with Multiple Languages

About Uninstalling Siebel CRM

Related Books

Siebel Bookshelf for each applicable prior release

Some earlier versions of the *Siebel Installation Guides* for the operating system you are using on My Oracle Support.

10 Requirements for Installing and Configuring Siebel CRM

Requirements for Installing and Configuring Siebel CRM

This chapter documents requirements that apply to installing and configuring Siebel CRM software. It includes the following topics:

- *About Installing Siebel CRM*
- *Planning Your Siebel Deployment*
- *Requirements for Installing and Configuring Siebel Enterprise Components*
- *Requirements for Installing and Configuring the Siebel Application Interface*
- *Requirements for Installing and Configuring the Siebel Database*
- *Requirements for Installing Siebel Web Clients*
- *Requirements for Installing Siebel Tools*

About Installing Siebel CRM

Each Siebel CRM software release from Oracle has a specific version level, such as Siebel CRM 22.7. You can install Siebel Enterprise Components (including Siebel Server, Siebel Gateway, Siebel Application Interface, and other components), Siebel Web Client, Siebel Tools, and other components at the current version. This topic includes the following information:

- *About Documented Requirements*
- *About Siebel CRM Releases*
- *About Installing in Upgrade Environments*
- *About Language Installation*

Related Topics

Siebel CRM Download and Installation

Uninstalling Siebel CRM

About Documented Requirements

This topic is part of *About Installing Siebel CRM*.

Before installing any Siebel CRM release, observe all of the documented requirements and review documented installation tasks. See this guide or any updated version of it and any other relevant documents on Oracle Help Center, Oracle Technology Network, or My Oracle Support. See also *Siebel CRM Update Guide and Release Notes* on My Oracle Support.

Note: All Siebel CRM modules used within a given Siebel Enterprise must share the same release level (version number), such as Siebel CRM 22.1. Third-party products must use supported release levels as documented in *How to Check the Certification Matrix in the New My Oracle Support Site* (KB871388) on My Oracle Support.

Examples of installation requirements follow:

- Before installing Siebel CRM for Siebel Enterprise Server (in a migration installation case), you must shut down services for Siebel Server or Siebel Gateway Name Server, and shut down any running instances of srvrmgr. For *Siebel Installation Guides* for some prior releases, see My Oracle Support.
- Before installing Siebel CRM for Siebel Application Interface (in a migration installation case), you must stop the Web server used for the previously installed release. For *Siebel Installation Guides* for some prior releases, see My Oracle Support.

About Siebel CRM Releases

This topic is part of *About Installing Siebel CRM*.

Siebel CRM software releases and their various installation methods have evolved significantly. Some previous types of releases no longer apply in the current release of Siebel CRM. This book describes the tasks and requirements that apply for the current release.

This guide, *Siebel Installation Guide* , is for installing the current release. You install Siebel Enterprise Components, Siebel Web Client, or Siebel Tools as a new installation of the software, as an update of an existing installation of Siebel CRM 17.0 or later, or as a migration installation of an existing installation of Siebel CRM 16.x or earlier. For more information, see *Installation-Related Changes in Siebel CRM* and related topics.

Note: For an existing installation of a prior version (before Siebel CRM 17.0), running Incremental Repository Merge is required after performing a migration installation. For more information, see *About Database Updates for Siebel CRM*. Other requirements apply. See also *Additional Tasks for Migration Installations*.

The following table lists the major installable Siebel CRM modules for Siebel CRM and shows which installation methods are supported in order to install the current release. Both new installations and migration installations of prior releases (before Siebel CRM 17.0) use the same installers.

Siebel CRM Module	Installation Methods for Current Release of Siebel CRM
Siebel Enterprise Components (including Siebel Gateway, Siebel Server, Siebel Application Interface, other components) Siebel Web Client Siebel Tools	• New installation of Siebel CRM, as described in *Installing Siebel CRM in a New Installation*. • Installation of Siebel CRM to update an existing installation of Siebel CRM 17.x or later, as described in *Installing Siebel CRM in an Update Installation*. • Migration installation of Siebel CRM, as described in *Installing Siebel CRM in a Migration Installation*.

Few additional Siebel application configuration tasks are associated with a migration installation. You would not have to run the Siebel Management Console (successor to Siebel Configuration Wizards) to configure a module that was already configured. However, requirements vary for different releases, or for using particular functionality. For example, after a migration installation you must configure the Siebel Application Interface. In some cases, you might also need to reconfigure a server module.

Note: The installation tasks and requirements for all future releases are subject to change. You must verify the availability, characteristics, and requirements of any future Siebel CRM releases that might also be covered by this guide (or updated versions of it) or by other documentation.

Related Books

For the *Siebel Installation Guide* for your operating system, for releases through Siebel CRM 21.1, see *Siebel Bookshelf* for each applicable release. For some prior releases, see My Oracle Support.

See also *Siebel CRM Update Guide and Release Notes* on My Oracle Support for each applicable release.

About Installing in Upgrade Environments

This topic is part of *About Installing Siebel CRM*.

When you install Siebel CRM, you might be creating a new deployment or you might be installing updated software to be used in a deployment for which the Siebel database is being upgraded from a previous release of Siebel CRM, such as an upgrade from Siebel CRM 8.0 to the current version. A deployment that's being upgraded is sometimes called an *upgrade environment*. You install your new Siebel CRM software before upgrading the Siebel database.

Sometimes, you can upgrade the Siebel database directly to the current release. For detailed information about upgrading the Siebel database, including upgrade paths, and about tasks that you perform before the installation or upgrade, see *Siebel Database Upgrade Guide* . See also any relevant documents on Oracle Technology Network or My Oracle Support.

Note: Review all of the applicable documentation before you install or upgrade.

For IBM DB2 for z/OS, see relevant parts of *Planning RDBMS Installation and Configuration* and see *Siebel Database Upgrade Guide for DB2 for z/OS* .

CAUTION: As part of installing Siebel CRM software in an upgrade environment, in general, you don't perform any database-related tasks described in this guide.

In an upgrade environment:

- You *do install* the Siebel Enterprise Server module called *Database Configuration Utilities* on a Siebel Server, as described in *Siebel CRM Download and Installation*. This module is always included in installations of Siebel Enterprise Components.
- You *don't create* the database instance on the RDBMS, as described in *Configuring the RDBMS*. (After the upgrade is complete, you might need to change database parameters for your RDBMS platform to match settings described in that chapter.)
- You *don't perform* the tasks described in *Installing the Siebel Database on the RDBMS*.
- Certain installation and configuration requirements associated with authentication for using Siebel Gateway and Siebel Management Console that apply for new deployments don't apply in this case. For details, see *Requirements for Siebel Gateway Authentication*.

For more information about how to use this guide in an upgrade installation case, see *Roadmap: Installing Siebel CRM in an Upgrade Case (Existing Database Requiring Full Upgrade)*.

After installing and upgrading, you might need to perform additional tasks. Some tasks that might apply are described in *Siebel Application Deployment Manager Guide* , *Siebel System Administration Guide* , or *Siebel Database Upgrade Guide* .

As of Siebel CRM 17.0, the Siebel Migration application, a Web-based tool for migrating Siebel Repositories and seed data and performing related tasks, is provided with the Siebel Application Interface installation. (Both Siebel Application Interface and Siebel Migration are always included in installations of Siebel Enterprise Components.) The Siebel Management Console includes tasks for configuring the Siebel Migration application. *Siebel Database Upgrade Guide* describes the Siebel Migration application. See also *Preparing to Run Siebel Server Components*.

When you install Siebel CRM in some upgrade scenarios, you might in some cases be able to install different versions of a Siebel CRM module on the same computer or operating system instance, if this scenario is valid for particular modules and might be helpful to you. For more information, see *Planning to Install Multiple Instances of Siebel CRM*.

Install the current release of Siebel CRM *before* you initiate the upgrade process. After upgrading, you can install additional Siebel CRM releases as they become available.

CAUTION: Don't install any additional Siebel CRM release until the upgrade is complete.

For information about types of Siebel CRM releases, see *About Siebel CRM Releases*.

Instructions in this guide apply both for new installations and upgrades, except where noted.

About Language Installation

This topic is part of *About Installing Siebel CRM*.

It is strongly recommended that you install and deploy, or at least install, all of the languages that you expect to require. For more information about installing and deploying languages, see *Installing and Deploying Siebel CRM with Multiple Languages*.

Planning Your Siebel Deployment

You must plan where to install the various Siebel CRM modules on your servers. This topic includes the following information:

- *Preparing the Hardware and Planning Topologies*
- *Planning Disk Space and Configuration Requirements*
- *Planning RDBMS Installation and Configuration*
- *Managing Temporary Disk Space Required by Siebel Installers and Wizards*
- *About the Language in Which Siebel Installers and Wizards Run*
- *Planning to Install Multiple Instances of Siebel CRM*
- *Specifying the Locale for Siebel CRM*
- (UNIX) *Specifying the Locale for Your UNIX Operating System*
- (UNIX) *Setting Permissions and Ownership on UNIX*

- (UNIX) *Installing Siebel CRM from a Remote Server Computer on UNIX*
- *Restrictions on Host Names for Siebel Gateway and Siebel Server*
- *Restrictions on Names for Siebel Enterprise and Siebel Server*
- *File and Directory Naming Conventions*
- *Creating the Siebel File System*
- *Creating the Siebel Service Owner Account*

Preparing the Hardware and Planning Topologies

This task is part of *Planning Your Siebel Deployment*.

Verify that the hardware that you have chosen meets all of the requirements for running your Siebel applications as well as the required third-party software. Verify also that the hardware is able to support the RDBMS and the Siebel database, the Siebel File System, Siebel Gateway, Siebel Server, Siebel Application Interface, and other Siebel CRM modules. Also plan to support Siebel Tools developer workstation computers and Siebel Mobile Web Client computers, where applicable, as well as mobile devices if you are deploying Siebel Mobile applications.

Consider running specialized components on dedicated computers. Whether you do so depends on considerations that include available resources, overall load, performance, security, and so on.

In general, it is assumed that, for security purposes, your computer resources are not shared between Siebel CRM and other applications or systems that are not related to supporting your Siebel CRM deployment.

Record the computer names and installation directory names that you decide on in your copy of the worksheet in *Siebel Deployment Planning Worksheet*.

Related Topics

Planning to Install Multiple Instances of Siebel CRM

Restrictions on Host Names for Siebel Gateway and Siebel Server

Restrictions on Names for Siebel Enterprise and Siebel Server

File and Directory Naming Conventions

Creating the Siebel File System

About the Siebel Application Interface

Planning the Siebel Application Interface Topology

Related Books

Siebel Deployment Planning Guide

Siebel Performance Tuning Guide

Siebel Security Guide

Planning Disk Space and Configuration Requirements

This task is part of *Planning Your Siebel Deployment*.

Before you install, you must anticipate your disk space requirements for each installable Siebel CRM module. Each Siebel installer displays the required disk space for the installed software before files are copied. Also anticipate the disk space required for Siebel image directories.

Siebel installers also have temporary disk space requirements, as described in *Managing Temporary Disk Space Required by Siebel Installers and Wizards*.

The Siebel CRM architecture is designed to scale across multiple application server computers to meet the needs of large, complex deployments. For assistance in capacity planning and sizing hardware platforms for Siebel CRM server modules, contact your Oracle sales representative or Oracle Advanced Customer Services to request assistance from Oracle's Application Expert Services.

The number of server computers, the CPU resources, and the memory resources required for operating Siebel CRM increases with the number and types of users and the deployed Siebel CRM modules.

In addition to the disk space required for installation, computers operating Siebel Remote components require approximately 15 MB of usable disk space for each mobile user. Oracle recommends using high-performance and high-availability storage solutions, as appropriate, for Siebel CRM deployments, particularly for high-capacity deployments. Oracle also recommends using high-performance disks or fault-tolerant disk arrays for all server computers, Siebel File System servers, and database servers.

Hardware or software high-availability disk configurations are critical for Siebel Remote servers, Siebel File System servers, and database servers. It is recommended that you configure other Siebel Server computers, Siebel Application Interface computers, and other components to use hardware or software RAID level 1 or comparable configurations.

High-availability solutions, such as clustering, are also recommended on critical components and can be used in conjunction with applicable high-availability storage solutions. Clustering is critical for high availability of the Siebel Gateway, Siebel Remote servers, and the database server. Clustering is also recommended for any Siebel Server computer that is not load-balanced.

Customers are responsible for making sure that clustering is supported on each server component and that the Siebel Server hardware operates with the supported server operating systems and clustering solutions.

Platform, database, and other certifications for Siebel are found on My Oracle Support. For help searching for certifications, see *How to Check the Certification Matrix in the New My Oracle Support Site* (KB871388).

It is strongly recommended to deploy Siebel Gateway clustering on multiple computers, as described in *Configuring the Siebel Gateway Cluster*.

For more information about some of these recommendations, see *Siebel Deployment Planning Guide* . See also *Siebel Performance Tuning Guide* .

Planning RDBMS Installation and Configuration

This task is part of *Planning Your Siebel Deployment*.

Note the following guidelines for installing and configuring the Relational Database Management System (RDBMS) on which you will create the Siebel database.

For more information about configuring the RDBMS, see *Configuring the RDBMS*. For more information about configuring the Siebel database after installing Siebel Enterprise Server, see *Installing the Siebel Database on the RDBMS*.

- Make sure that this release of Siebel CRM supports the exact version of your chosen RDBMS and that the RDBMS has been installed on its designated server. This database server will hold the database tables containing your business data, such as sales (personnel, territories, opportunities, and activities), marketing, and customer service information. It will also contain Siebel Repository tables, which are used both for developing and for operating Siebel applications.

  Platform, database, and other certifications for Siebel are found on My Oracle Support. For help searching for certifications, see *How to Check the Certification Matrix in the New My Oracle Support Site* (KB871388).
- Verify that the network name of the server that will support the Siebel database is properly recorded in your copy of the worksheet in *Siebel Deployment Planning Worksheet*.
- Installing the Siebel database, as described in *Installing the Siebel Database*, creates a ODBC data source name, which might be needed to provide authentication for the Siebel Management Console configuration process, as described in *Configuring Siebel CRM Server Modules*.
- When you configure the Siebel Enterprise, the Siebel Management Console creates the ODBC data source name, using the name SiebelEnterpriseName_DSN. For example, if your Siebel Enterprise name is sieb21, then the ODBC data source name will be sieb21_DSN. If a data source using this name exists, then it is overwritten. Using this pattern, determine what your ODBC data source name will be and record it in your copy of the worksheet in *Siebel Deployment Planning Worksheet*.

  > **Note:** In most cases, do not change any of the default ODBC settings or parameters. Otherwise, you will encounter problems using the ODBC. For example, setting OptimizeLongPerformance to `enable` corrupts all of the scripts in the Siebel Repository during import or export processes. For information about verifying the ODBC data source for your RDBMS platform, see *Verifying the ODBC Data Source*. See also *Troubleshooting the ODBC Data Source Connection*.

  Siebel Web Client and Siebel Tools installers also create ODBC data source names during the installation. The naming convention is different than it is for ODBC data source names created by the Siebel Management Console. For more information, see *Installing Siebel Web Clients and Siebel Tools*.

About Database Code Pages and Locale Support

This topic is part of *Planning RDBMS Installation and Configuration*.

In a Unicode database environment, you can install any of the available Siebel languages.

In a database environment that is not enabled for Unicode, however, you must consider the correlation of the languages that you want to install and the characters supported by your database code page. For example, in a Western European code page database, you can only install Western European languages, such as U.S. English, French, Spanish, or German languages. In a Japanese code page database, you can only install Japanese or U.S. English languages. The installed languages and the code page of the operating system on which your Siebel Server runs must match those of your Siebel database instance.

Related Topics

Specifying the Locale for Siebel CRM

(UNIX) *Specifying the Locale for Your UNIX Operating System*

Configuring the RDBMS

Related Books

Siebel Global Deployment Guide

Planning Database Connectivity

This topic is part of *Planning RDBMS Installation and Configuration*.

Make sure that ODBC connectivity uses TCP/IP as the transport layer protocol for Siebel Servers, Siebel Tools, and Siebel Developer Web Clients.

Platform, database, and other certifications for Siebel are found on My Oracle Support. For help searching for certifications, see *How to Check the Certification Matrix in the New My Oracle Support Site* (KB871388).

How to Use This Guide If Your Database Is IBM DB2 for z/OS

This topic is part of *Planning RDBMS Installation and Configuration*.

This guide describes installation of Siebel CRM modules for each supported RDBMS, except for IBM DB2 for z/OS. (This database product was formerly referred to in some *Siebel Bookshelf* documentation as *IBM DB2 UDB for z/OS and OS/390*.)

Customers for IBM DB2 for z/OS require this guide (*Siebel Installation Guide*) and *Implementing Siebel Business Applications on DB2 for z/OS* . Both of these guides are available on the *Siebel Bookshelf* .

> **Note:** In this guide, *Configuring the RDBMS* and *Installing the Siebel Database on the RDBMS* don't apply to customers using DB2 for z/OS. Instead, see *Implementing Siebel Business Applications on DB2 for z/OS* for database-related content. If you're upgrading on this platform, then see also *Siebel Database Upgrade Guide for DB2 for z/OS* . For information about relevant requirements related to authentication for the Siebel Gateway, see *Requirements for Siebel Gateway Authentication*.

Implementing Siebel Business Applications on DB2 for z/OS includes, but isn't limited to, topics such as special security issues for this RDBMS, installing the Siebel schema, performance guidelines for use of this RDBMS with Siebel CRM, and specific procedures for moving data from development to production. Also consult all of the other applicable *Siebel Bookshelf* documentation, such as *Siebel Security Guide* .

Managing Temporary Disk Space Required by Siebel Installers and Wizards

This task is part of *Planning Your Siebel Deployment*.

The Siebel CRM installer and other programs all require the use of temporary disk space while operating. Disk space required varies by operating system. The location of temporary directories is specified using the TEMP and TMP environment variables. Set these variables the same: to a valid existing directory that contains adequate space.

It is recommended that you periodically clear out the temporary directory that you are using. You might have to do this as part of resolving installer or wizard problems.

On Microsoft Windows, the temporary directory used by default is `C:\Documents and Settings\username\Local Settings\Temp`.

The minimum disk space figures provided in this topic are general guidelines only. The specific amount of temporary disk space needed by a particular installation operation varies. If there is insufficient disk space, then the installer will notify you that more disk space is needed.

> **CAUTION:** (Windows) A nondefault temporary directory name that you specify must not include spaces, in order to avoid problems associated with how Microsoft Windows handles long path names. Installers or wizards might fail if this problem is encountered.

The user running the installer or wizard must have privileges that include write permission for the temporary directory.

The following shows the minimum disk space required by Siebel installers and wizards (by operating system), along with the default temporary directory locations.

Operating System	Minimum Required Space	Default Location
Windows	65 to 100 MB	System drive (usually `C:`)
AIX	200 MB	`/tmp`
HP-UX	240 MB	`/var/tmp`
Linux	200 MB	Home directory for user
Oracle Solaris	115 MB	`/var/tmp`

To verify the name of your temporary directory

- From a DOS command prompt, enter the commands `set temp` and `set tmp`.

About the Language in Which Siebel Installers and Wizards Run

This topic is part of *Planning Your Siebel Deployment*.

When you start the Siebel CRM installer or the Siebel Database Configuration Wizard, the language in which the program runs (the user interface text) is determined by the language for the locale of your operating system, where this language and locale are supported for Siebel CRM and for the installer, utility, or wizard.

Installers for Siebel CRM server modules and client modules are based on Oracle Universal Installer and run in the current language corresponding to the regional and language options settings on the server or client computer, where the current language is one of the following: U.S. English, French, German, Italian, Brazilian Portuguese, Spanish, Japanese, Korean, Chinese Simplified, or Chinese Traditional. For any other current language setting, the installer runs in U.S. English.

The Siebel Database Configuration Wizard also uses the current language corresponding to the regional and language options setting. The program can run in all of the languages corresponding to the available Siebel languages, except Arabic, Hebrew, Polish, Russian, Thai, or Turkish. If the current language is one of these or any language not supported for Siebel CRM, then the program runs in U.S. English.

For the Siebel CRM installer and the Database Configuration Wizard, in order to change the user interface language, you must change the current operating system language and locale. Make sure that the language setting under the regional option of your operating system is set to the desired choice of your language. The language that you select must be one of those supported for Siebel CRM, as described earlier.

(UNIX) For the Siebel CRM installer and the Database Configuration Wizard, in order to change the user interface language, you must change the current operating system language and locale. Make sure that the locale of your operating system is set to the desired choice of your language. The language that you select must be one of those supported for Siebel CRM, as described earlier.

> **Note:** (UNIX) If a supported locale that you intend to use does not exist on the supported operating system, then use another existing supported locale, so that you can proceed with installation and initial configuration in a familiar language; for example, DE_DE.UTF-8 for German (on AIX), and so on. For a list of supported locales for your UNIX operating system, see *Specifying the Locale for Your UNIX Operating System*.

For information about installing languages for Siebel CRM, see the applicable installation task topic and see *Installing and Deploying Siebel CRM with Multiple Languages*.

Planning to Install Multiple Instances of Siebel CRM

This task is part of *Planning Your Siebel Deployment*.

This topic describes issues in installing multiple instances of similar or dissimilar versions of Siebel CRM on the same computer or operating system instance.

This topic includes the following information:

- *Installing Multiple Instances of the Same Version of Siebel CRM*
- *Installing Multiple Instances of Different Versions of Siebel CRM*

Installing Multiple Instances of the Same Version of Siebel CRM

This task is part of *Planning to Install Multiple Instances of Siebel CRM*.

By default, the Siebel CRM installer automatically installs Siebel CRM software in directories with unique names that will not conflict with other Siebel CRM modules that you install on the same computer or operating system instance. To install another instance of the same version of a module on a single computer or operating system instance (where applicable), specify a custom installation path.

However, installing multiple instances of some modules might cause problems. For example:

- (Windows) You cannot install multiple instances of the same version (or any version) of Siebel Gateway on the same computer or operating system instance. The reason for this is that installing and configuring the Siebel Gateway creates a service which cannot coexist in multiple instances for different installations.
- (UNIX) You cannot install multiple instances of the same version of Siebel Gateway on the same computer or operating system instance.

- You must take care to avoid conflicts in port number usage whenever you install or configure multiple Siebel CRM modules or multiple instances on the same computer or operating system instance. You specify multiple port numbers for application containers when you install and configure Siebel CRM modules.

For important related considerations, see also *About Installing Siebel CRM* and *Preparing the Hardware and Planning Topologies*.

Note: In this context, *same version* refers to multiple installations of a given release, such as Siebel CRM 22.7.

All Siebel CRM server modules that you install together on the same computer or operating system instance are installed into the same root directory (such as `c:\siebel` on Microsoft Windows or `/siebel` on UNIX). Siebel Enterprise Components you install include Siebel Gateway, Siebel Server, Siebel Application Interface Database Configuration Utilities, and EAI Connector. Restrictions and guidelines apply to collocating Siebel CRM modules, as noted in *About Installing Siebel CRM*. See also *File and Directory Naming Conventions*.

If you have already installed Siebel CRM modules (Siebel Enterprise Server) for the current release, and you require additional Siebel CRM modules (for example, Siebel Gateway is already installed and you want to install a Siebel Server), then you can install a new instance of Siebel Enterprise Components. You must install the new instance in a different directory or on a different computer or operating system instance. For more information about options in this situation, see *General Requirements for Installing and Configuring Siebel Enterprise Components* and other relevant topics.

Each installed instance of Siebel CRM software, whether on the same or on a different computer or operating system instance, must be configured in order to operate it in your Siebel deployment. Configuration requirements are different for new installations and migration installations. A new installation must be configured using Siebel Management Console, by configuring and applying profiles for each module. Depending on your existing deployment, in a migration installation case involving multiple instances of a Siebel CRM module, you might have a choice about whether to install and configure each additional instance of a new module as a new installation or as a migration installation. Specific requirements vary by Siebel CRM module and by installation case.

If you install additional instances of Siebel Application Interface (when the primary instance is operational, whether a new or migration installation), then some additional steps might be necessary to deploy these instances. These steps involve editing the applicationinterface.properties file, which is located in `SIEBEL_AI_ROOT/applicationcontainer_external/webapps`. Copy the following parameters and their values from applicationinterface.properties on the primary instance of Siebel Application Interface into the [Application Interface] section of the corresponding file on the new instance: Password, UserName, and CGHostURI. For each new Siebel Application Interface instance, after you save the updated file, restart the application container. Then you can apply a profile to deploy this instance.

Installing Multiple Instances of Different Versions of Siebel CRM

This task is part of *Planning to Install Multiple Instances of Siebel CRM*.

Multiple versions of some Siebel CRM modules can be installed on the same computer or operating system instance. Sometimes you might have to do this as part of some upgrade scenarios.

- (Windows) Each instance of Siebel Gateway must be installed on a separate computer or operating system instance, regardless of the version.
- If you install multiple versions of Siebel Enterprise Components on the same computer or operating system instance, then you must install them in different directories and assign unique port numbers for the application containers.

Related Topics

Siebel CRM Download and Installation

Requirements for Installing and Configuring the Siebel Application Interface

Related Books

Siebel Database Upgrade Guide

Specifying the Locale for Siebel CRM

This task is part of *Planning Your Siebel Deployment*.

When a Siebel application component (Siebel Server, Siebel Tools, or Siebel Developer Web Client) opens a connection to the Siebel database, the locale is automatically determined on a per-session basis for the connection. This setting overrides settings that might be defined elsewhere.

(Windows) Do not explicitly set the NLS_LANG (Oracle-specific) or SIEBEL_CODEPAGE environment variables for Siebel CRM. Windows Registry settings for these variables might be defined, but these settings are not used for the database connection. These registry settings affect any software that runs on that computer (except for Siebel CRM).

(UNIX) Do not explicitly set the NLS_LANG (Oracle-specific), LANG, or SIEBEL_CODEPAGE environment variables for Siebel CRM. Siebel environment setup files such as siebenv.csh (for C shell) or siebenv.sh (for Bourne or Korn shell) might set these variables, but these settings are not used for the database connection. The settings in the siebenv.csh or siebenv.sh files affect any software that runs in the Siebel environment controlled by those files (except for Siebel CRM).

For deployments using Oracle Database, the NLS_LANG variable determines default behaviors that can be overridden using other variables, such as NLS_SORT, that are *not* set automatically for Siebel CRM using an Oracle Database connection. Therefore, settings that you make for NLS_SORT will affect any software that runs in the Siebel environment (*including* Siebel CRM).

> **CAUTION:** For development environments, Siebel CRM supports only binary sort order. Therefore, for Oracle Database, when you are setting the locale for your development environment, either do not set NLS_SORT at all, or set it to `BINARY`. In production environments, this restriction does not apply. (In this guide, *sort order* and *collation sequence* are used interchangeably, even though these terms might not always mean the same thing.)

You can explicitly set the locale to be used by a Siebel Server or Application Object Manager component by setting the Locale Code parameter.

Related Topics

Planning RDBMS Installation and Configuration

(UNIX) *Specifying the Locale for Your UNIX Operating System*

Configuring the RDBMS

Related Books

Siebel Applications Administration Guide

Siebel Global Deployment Guide

Specifying the Locale for Your UNIX Operating System

This task is part of *Planning Your Siebel Deployment*.

To successfully run Siebel CRM on UNIX, you must configure the appropriate locales on your operating system for the language that you will run the applications in. In some cases, you might have to install additional content on your system in order to support the locale.

For LANG or LC_ALL settings on different UNIX operating systems for supported languages or locales, see the information that follows.

Note: The setting of the LANG environment variable does not affect database connections for Application Object Manager components for Siebel CRM. However, it does affect database connections used by other Siebel components.

Language or Locale Siebel Language Code	LANG or LC_ALL Parameter Setting for Each Supported UNIX Operating System		
	AIX	HP-UX or Linux	Oracle Solaris
Arabic (Saudi Arabia) ARA	AR_SA.UTF-8	ar_SA.utf8	ar_SA.UTF-8
Chinese (Simplified) CHS	ZH_CN.UTF-8	zh_CN.utf8	zh_CN.UTF-8
Chinese (Traditional) CHT	ZH_TW.UTF-8	zh_TW.utf8	zh_TW.UTF-8
Czech CSY	CS_CZ.UTF-8	cs_CZ.utf8	cs_CZ.UTF-8
Danish DAN	DA_DK.UTF-8	da_DK.utf8	da_DK.UTF-8
Dutch NLD	NL_NL.UTF-8	nl_NL.utf8	nl_NL.UTF-8
English (United States) ENU	EN_US.UTF-8	en_US.utf8	en_US.UTF-8
Finnish FIN	FI_FI.UTF-8	fi_FI.utf8	fi_FI.UTF-8
French (France) FRA	FR_FR.UTF-8	fr_FR.utf8	fr_FR.UTF-8
German (Germany) DEU	DE_DE.UTF-8	de_DE.utf8	de_DE.UTF-8
Hebrew HEB	IW_IL.UTF-8	iw_IL.utf8	iw_IL.UTF-8
Italian ITA	IT_IT.UTF-8	it_IT.utf8	it_IT.UTF-8

Language or Locale Siebel Language Code	LANG or LC_ALL Parameter Setting for Each Supported UNIX Operating System		
	AIX	HP-UX or Linux	Oracle Solaris
Japanese JPN	JA_JP.UTF-8	ja_JP.utf8	ja_JP.UTF-8
Korean KOR	KO_KR.UTF-8	ko_KR.utf8	ko_KR.UTF-8
Polish PLK	PL_PL.UTF-8	pl_PL.utf8	pl_PL.UTF-8
Portuguese (Brazil) PTB	PT_BR.UTF-8	pt_BR.utf8	pt_BR.UTF-8
Portuguese (Portugal) PTG	PT_PT.UTF-8	pt_PT.utf8	pt_PT.UTF-8
Russian RUS	RU_RU.UTF-8	ru_RU.utf8	ru_RU.UTF-8
Spanish (Modern) ESN	ES_ES.UTF-8	es_ES.utf8	es_ES.UTF-8
Swedish SVE	SV_SE.UTF-8	sv_SE.utf8	sv_SE.UTF-8
Thai THA	TH_TH.UTF-8	th_TH.utf8	th_TH.UTF-8
Turkish TRK	TR_TR.UTF-8	tr_TR.utf8	tr_TR.UTF-8

Related Topics

Specifying the Locale for Siebel CRM

Managing Environment Variables on UNIX

Setting Permissions and Ownership on UNIX

This task is part of *Planning Your Siebel Deployment*.

This topic provides information about permissions and ownership for user accounts that will perform installations of the current release of Siebel CRM.

- To install server-based Siebel CRM modules, you must be a nonroot user. You cannot install these modules as the root user using Oracle Universal Installer software. This requirement applies to all installations of Siebel Enterprise Server and Siebel Application Interface.

  For existing installations of Siebel CRM 8.1.1.7 or earlier that were originally installed as the root user (using InstallShield installers), you can use the `chown` command to change the owner for all of the installation directories and files to a nonroot user in order to perform the migration installation.

  After installing as a nonroot user, you can optionally change the installation owner back to the root user. Note that the same requirement will apply every time that you install a new release of Siebel CRM. In general, it is recommended not to change the owner back to the root user, unless your deployment requires it.
- Before you install each server module discussed in this guide, set `umask` to `027` on the installation directory of that computer. Changing this setting eliminates other permissions, including group write permissions. This action also sets the default permissions, so that all of the files and directories that are created afterwards have `rwxr-x---` permissions. The Siebel service owner account must have all permissions. However, the group must have read and execute (for directory access) permissions only. See also *Creating the Siebel Service Owner Account*.

Related Topics

(UNIX) *Configuring the Siebel Gateway for Automatic Start on UNIX*

(UNIX) *Configuring the Siebel Server for Automatic Start on UNIX*

Installing Siebel CRM from a Remote Server Computer on UNIX

This task is part of *Planning Your Siebel Deployment*.

If you are installing server-based Siebel modules in GUI mode from a remote server computer, then you must set the DISPLAY variable to display the Java installer user interface on your local computer. For example, depending on your shell:

```
export DISPLAY=mycomputer:0.0
```

or:

```
setenv DISPLAY mycomputer:0.0
```

In this command, mycomputer is the computer name or IP address of your local workstation computer.

You might be able to test that your display works correctly on your local computer by entering:

```
xclock
```

You can also verify the IP address by entering:

```
echo $DISPLAY
```

If the clock does not appear on your local computer, then issue the following command on your local computer:

```
xhost +
```

If you are using X-connectivity software to access the UNIX computer from a Windows computer, then the installer might fail or stop responding. If this is the case, then try using a later version of the X-connectivity software.

> **Note:** For additional information, refer to vendor or other documentation for your UNIX operating system, or contact your system administrator for assistance.

Restrictions on Host Names for Siebel Gateway and Siebel Server

This topic is part of *Planning Your Siebel Deployment*.

See also *Restrictions on Names for Siebel Enterprise and Siebel Server*, which is about names for Siebel entities that you specify during configuration.

This topic describes restrictions for host names for server computers on which you will install Siebel Enterprise Components (including Siebel Gateway and Siebel Server).

The Siebel Gateway name is defined automatically based on the host name of the server computer. However, problems might arise during configuration of Siebel CRM modules unless the following requirements are observed:

- The host name for Siebel Gateway must be no longer than 15 characters.

  The Siebel Management Console task for configuring the Siebel Gateway displays an error if the host name you enter is longer than 15 characters.

  (Siebel Management Console tasks for configuring the Siebel Enterprise and the Siebel Server also validate that the name of the Siebel Gateway, which would have been previously configured, is no longer than 15 characters.)
- In general, it is recommended not to define the host name for the Siebel Gateway computer to include domain information, as might be common in some UNIX environments. The entire host name is subject to the 15-character limit mentioned previously.

Restrictions on Names for Siebel Enterprise and Siebel Server

This topic is part of *Planning Your Siebel Deployment*.

This topic describes restrictions to observe when planning the names that you will give the Siebel Enterprise and Siebel Server instances during configuration of Siebel CRM software.

See also *Restrictions on Host Names for Siebel Gateway and Siebel Server*.

Siebel Management Console tasks that specify or refer to the Siebel Enterprise name or the Siebel Server name validate for some of the following requirements, such as the 12-character limit. User input is validated in the Siebel Management Console. Observe all of the documented restrictions regardless of validation behavior.

This topic includes the following information:

- *Siebel Enterprise Naming Restrictions*
- *Siebel Server Naming Restrictions*

Siebel Enterprise Naming Restrictions

Here are the restrictions that apply to naming a Siebel Enterprise:

- The name must be no longer than 12 characters.
- The name can't be `server` or `enterprise`. (A name like `enterprise1` is acceptable.)

Related Topics

- Siebel Server Naming Restrictions

Siebel Server Naming Restrictions

Here are the restrictions that apply to names of Siebel Server instances within a Siebel Enterprise:

- Names must contain only alphabetic characters, numerals, underscores, dashes (hyphens), or a combination thereof.
- Names must lead with an alphabetic character.
- Names must be unique within the Siebel Enterprise.
- Names must be no longer than 12 characters.
- Names can't be `server` or `enterprise`. (A name like `enterprise1` is acceptable.)

Related Topics

- Siebel Enterprise Naming Restrictions

File and Directory Naming Conventions

This topic is part of *Planning Your Siebel Deployment*.

Use lowercase for all of the file names, directory names, path names, parameters, flags, and command-line commands, unless you are instructed otherwise.

Directory names or file names cannot contain special characters, including periods, apostrophes, accents, number (pound or hash) signs, ampersands, or spaces (on UNIX). Underscores are acceptable. Spaces are not allowed for installations on Windows (though the operating system might otherwise allow it).

How This Guide Refers to Your Installation Directories

This guide uses the following conventions to refer to the installation directories either created by the installers or to which users navigate to access files and executable programs.

> **Note:** The default installation directories shown in this guide for Siebel Enterprise Components (including Siebel Application Interface), Siebel Web Clients, and Siebel Tools are those for new installations of the current release. All components can be installed together, or installed separately. Server modules are installed together as Siebel Enterprise Components and are not available as individual selectable options in the Siebel CRM installer. For migration and update installations from releases prior to Siebel CRM 21.2, your installation directories reflect the products originally installed into them. However, after installation your server installations now contain all server modules, including some that might not have been installed and configured before.

Microsoft Windows

SIEBEL_ROOT. Generally, this refers to the main top-level directory in which you installed Siebel CRM software. The Siebel CRM installer installs software into the top-level directory, such as `c:\Siebel`, unless a different directory was specified. Generally, this directory is what `SIEBEL_ROOT` represents in this guide.

ORACLE_HOME. This refers to the same directory as represented by `SIEBEL_ROOT` .

SIEBEL_HOME. The installation or root directory for Siebel CRM software in general (such as `c:\Siebel`). However, depending on the context, this might refer to the root directory of a specific module such as Siebel Gateway or Siebel Server (such as `c:\Siebel\siebsrvr` for Siebel Server). Many scripts contain variables with this name; its meaning is most often derived from the context.

SIEBSRVR_ROOT. The root directory for Siebel Server, such as `c:\Siebel\siebsrvr`.

SIEBEL_GATEWAY_ROOT. The root directory for Siebel Gateway, such as `c:\Siebel\gtwysrvr`.

DBSRVR_ROOT. The root directory for Siebel Database Configuration Utilities (also called the *Siebel Database Server*), such as `c:\Siebel\dbsrvr` (on a Siebel Server computer).

SIEBEL_AI_ROOT. The root directory for Siebel Application Interface, such as `c:\Siebel\applicationinterface`.

SIEBEL_CLIENT_ROOT. The installation or root directory for Siebel Web Client, such as `c:\Siebel\Client` or a similar location.

SIEBEL_TOOLS_ROOT. The installation or root directory for Siebel Tools, such as `c:\Siebel\Tools` or a similar location.

UNIX

$SIEBEL_ROOT. Generally, this refers to the main top-level directory in which you installed Siebel CRM software. The Siebel CRM installer installs software into the top-level directory, such as `/Siebel`, unless a different directory was specified, such as `/export/home/Siebel`. Generally, this directory is what $SIEBEL_ROOT represents in this guide.

> **Note:** $SIEBEL_ROOT might also represent the value of the SIEBEL_ROOT environment variable, which usually corresponds to a module-specific installation directory, such as `/Siebel/siebsrvr` for Siebel Server.

$ORACLE_HOME. This refers to the same directory as represented by $SIEBEL_ROOT.

$SIEBEL_HOME. The installation or root directory for Siebel CRM software in general (such as `/Siebel`). However, depending on the context, this might refer to the root directory of a specific module such as Siebel Gateway or Siebel Server (such as `/Siebel/siebsrvr` for Siebel Server). Many scripts contain variables with this name; its meaning is most often derived from the context.

$SIEBEL_SERVER_ROOT. This term is sometimes used within executable programs, such as the `siebel_server` script, as a synonym for $SIEBEL_HOME.

$SIEBSRVR_ROOT. The root directory for Siebel Server, such as `/Siebel/siebsrvr`.

$SIEBEL_GATEWAY_ROOT. The root directory for Siebel Gateway, such as `/Siebel/gtwysrvr`.

DBSRVR_ROOT. The root directory for Siebel Database Configuration Utilities (also called the *Siebel Database Server*), such as `/Siebel/dbsrvr` (on a Siebel Server computer).

SIEBEL_AI_ROOT. The root directory for Siebel Application Interface, such as `/Siebel_AI`.

Creating the Siebel File System

This task is part of *Planning Your Siebel Deployment*.

The *Siebel File System* is a shared directory, or set of directories, that is network-accessible to the Siebel Server and that can store files such as attachments for use by Siebel CRM. Siebel File System directories might optionally exist on multiple separate devices or partitions.

Each File System directory might be created on a server computer where you have installed a Siebel Server, or on another network server that can share the directory, so that it is available to the Siebel Server. Consult your third-party documentation for requirements for networked file systems.

Creating multiple Siebel File System directories in different locations can enable you to store larger volumes of data. As new file attachments are inserted, they are evenly distributed to these multiple File System directories. If you create multiple File System directories, then you must include all of the directory locations, delimited by commas, when you specify the Siebel File System location during configuration of your Siebel environment. Each File System directory location must be uniquely named within the network context where it will be accessed.

You can exclude certain types of files from being saved to the Siebel File System, as determined by their file extensions. For more information about setting system preferences to enable this feature and to specify the file extensions to be excluded, see *Siebel Security Guide* .

This topic includes the following information:

- *Siebel File System Requirements*
- *Siebel File System and Siebel Server Components*
- *Siebel File System Parameter*
- *Deployments Using the Siebel Migration Application*
- *Siebel File System Utilities*
- *Naming a Siebel File System Directory*
- *Setting Up the Siebel File System*

Related Topics

Populating the Siebel File System

Related Books

Siebel Deployment Planning Guide

Siebel System Administration Guide

Siebel Security Guide

Siebel File System Requirements

The following are requirements for the Siebel File System. Additional requirements are noted later in this topic.

This topic is part of *Creating the Siebel File System*.

- You specify the Siebel File System directory when you configure and deploy the Siebel Enterprise using Siebel Management Console. Siebel Management Console creates the Siebel File System directory, if the directory

specified for the File System doesn't exist. However, it's recommended that you create the primary Siebel File System directory with appropriate permission before you configure the Siebel Enterprise.

> **Note:** Don't place the Siebel File System in a `$SIEBEL_HOME`. For example, if you install Siebel CRM to `D:\Siebel`, the File System can't be placed in `D:\Siebel\fs` or any other folder in that directory structure.

- The Siebel File System directory location must be specified using UNC format, such as `\\computer_name\FS`. If this directory is located on the same computer or operating system instance where you're installing and configuring Siebel CRM software, then the directory must be created as a shared directory.
- The user running the Siebel Management Console must have write permission in the Siebel File System directory.
- It's strongly recommended that you disable short file-name generation on Windows server computers or operating system instances hosting the Siebel File System. Using this type of file-naming can cause severe performance issues when the file system grows to a large size.
- The underlying physical file system must use file locking. Valid locks must be returned that are appropriate to the action being undertaken. For example, when a file is created or written to, an exclusive lock is needed. File locking must ensure the integrity of the files accessed. If a write lock is acquired for a file, then no other process must be able to access the file.
- If the operating systems of the computers hosting the Siebel Server and a File System directory are different (for example, one Windows and one UNIX), then you might need to deploy a third-party cross-platform networking tool, such as Samba, to allow both computers to share the directory. See your third-party supplier documentation for details.
- You must create a separate Siebel File System for each Siebel Enterprise. For example, if you have development and test databases, then you must have two separate Siebel Enterprises, and therefore two Siebel File Systems.
- The ability to use multiple directories and devices for the Siebel File System doesn't apply to the Siebel Mobile Web Client, for which the Siebel File System must use a single directory on the client computer.
- If you operate a File System directory as part of a cluster for failover purposes, then you must create the directory on a clustered disk drive with a clustered network share resource. For information about clustering your servers and about deployment options for the Siebel File System, see *Siebel Deployment Planning Guide* .
- The Siebel service owner account must have read-write access to the Siebel File System and the Migration Package Location. See also *Creating the Siebel Service Owner Account*.

Siebel File System and Siebel Server Components

This topic is part of *Creating the Siebel File System*.

Each Siebel Server accesses the Siebel File System (for the Siebel Enterprise) by means of a dedicated server component, called File System Manager (alias FSMSrvr). Individual Siebel Web Clients require no direct knowledge of the locations of the Siebel File System directories, because they connect to FSMSrvr through the Application Object Manager component on the Siebel Server to request file uploads or downloads. The Application Object Manager passes such requests to FSMSrvr, which processes the requests through interaction with the File System directories.

Because the Siebel Server is the sole access mechanism to the Siebel File System, the user with administrative privileges for the Siebel Server, and no other user, must have access privileges to the File System directories. This precaution protects the File System from direct physical access by all other users.

Some Siebel Server components might access the Siebel File System directly, without using File System Manager.

Siebel File System Parameter

This topic is part of *Creating the Siebel File System*.

The Siebel File System parameter defines the particular directory or set of directories that you are using for the Siebel File System. Specify multiple File System directories delimited by commas.

The Siebel File System parameter can be defined at the Siebel Enterprise level, Siebel Server level, or server component level. When you configure the Siebel Enterprise, you specify a value for this parameter, as described in *Creating a Siebel Enterprise Profile*.

For a Siebel Server dedicated for use with Siebel EIM, for example, you might choose to use a Siebel File System location that is not in general use in the Siebel Enterprise. Use Siebel Server Manager to individually modify the parameter at the Siebel Server or component level, if the File System that is to be used by a particular Siebel Server or applicable component has different directory locations than are defined for the Siebel Enterprise.

The value of the Siebel File System parameter cannot exceed 2048 characters in length.

> **Note:** Verify that the network names of servers that support the Siebel File System are properly recorded in your copy of the worksheet in *Siebel Deployment Planning Worksheet*. Use the computer names, not the IP addresses, for the Siebel File System names. IP addresses are not supported.

Deployments Using the Siebel Migration Application

This topic is part of *Creating the Siebel File System*.

Because the Siebel Migration application uses the Siebel File System, it is strongly recommended to use different file system locations for the source and target environments, to avoid any file conflicts. For this type of configuration, use Siebel Server Manager to set the following parameters in the source and target environments:

- Set the parameter DSFileSystem for the server data source (ServerDataSrc) to `*FSM*`.
- Set the parameter FileSystem for the enterprise to specify the file system locations applicable for that environment.

For more information, see the description of the Migration Package Location setting in *Configuring the Siebel Migration Application*. For more information about administering the Siebel File System and setting parameters, see *Siebel System Administration Guide* .

Siebel File System Utilities

This topic is part of *Creating the Siebel File System*.

The following utilities are available to help you manage your Siebel File System directories: `sfscleanup` and `sfspartition`.

- Use `sfscleanup` to clean up orphan files in the Siebel File System.
- Use `sfspartition` to configure an existing Siebel File System to use multiple partitions or multiple directories.

For information about using these utilities, see *Siebel System Administration Guide* .

Naming a Siebel File System Directory

This task is part of *Creating the Siebel File System*.

Each Siebel File System directory name must be alphanumeric, must begin with an alphabetic character, and cannot contain special characters or spaces. Underscores are permitted. For example, you might name a directory something like this:

(Windows) `\\Siebel_FS_Host\SiebelFS`

(UNIX) `/Siebel_FS_Host/SiebelFS` or `/Siebel_FS_Host/Siebel/Filesystem`

Such a directory might be referred to using the following notation:

(Windows) `\\Siebel_FS_Host\SiebelFS`

(UNIX) `Siebel_FS_Host/SiebelFS`

where:

- Siebel_FS_Host is the host name of the computer (for example, where a dedicated computer is used for the Siebel File System).
- SiebelFS is the name of the shared directory.

You have to specify all of the applicable UNC sharenames (delimited by commas) when configuring the Siebel Enterprise. These shared directories must be available to all of the Siebel Servers in the Siebel Enterprise.

Note: When you create a Siebel File System shared directory, only the associated Siebel Servers must be allowed to create subdirectories in that location. Do not manually create subdirectories in that location.

As part of the Siebel Server installation, File System Manager automatically generates a set of subdirectories under each Siebel File System root directory. Some of these directories are described in the following table.

Subdirectory	Purpose
`att`	Main subdirectory for file attachments
`atttmp`	Temporary directory for file attachments
`CFGCache`	Files used by Siebel Product Configurator
`CFGDefs`	Files used by Siebel Product Configurator
`cms`	Files used for outbound communication requests
`eim`	Siebel transaction files for Siebel EIM
`ISS_OBrkCache`	Files used by Siebel Product Configurator
`Marketing`	Main subdirectory for Siebel Marketing
`Migration`	Main subdirectory for the Siebel Migration application
`red`	Rule edit cache for Siebel Product Configurator
`ssp`	Session preferences
`userpref`	Siebel user preferences

For a migration installation case, Siebel File System subdirectories from previous releases are retained. Some of these directories might no longer apply in the current release, due to evolving feature support in Siebel CRM.

For more information about some of the Siebel File System subdirectories, see applicable documentation on *Siebel Bookshelf* , including *Siebel System Administration Guide* .

Setting Up the Siebel File System

This topic is part of *Creating the Siebel File System*.

Use the following procedures to set up the Siebel File System directories.

> **Note:** In addition to using the following steps, you must install the third-party software required to view standard attachment types, such as Microsoft Word or Excel, on client computers where users run Siebel CRM.

To set up the Siebel File System

1. Create each directory on the applicable server and record all of the directory locations in the copy that you made of the worksheet in *Siebel Deployment Planning Worksheet*.
2. Using Windows Explorer, select the directory, and then choose File, then Properties, and then Sharing. Select Share this folder.
3. Type a name for the Share name.

   > **Note:** When you want to add a client or new user to this share, you click Add under the Security tab to browse for the user or group name.

   You might want to type the number of the Siebel release that you are installing in the Comments field for future identification. However, completion of the field is not required.

   > **Note:** When installing a new File System directory, do not change the default setting for Maximum Allowed.

4. To grant UNC access to the Siebel administrator, click Permissions and choose the appropriate user or group name. When you want to add a client or new user to this share, you click Add to browse for the user or group name.

   > **Note:** Only the system administrator for the Siebel Server, and no other user, must have access privileges to the Siebel File System directories.

5. From the Permissions list, make sure that Full Control is selected.
6. Click OK to finish.
7. Grant Windows access to each Siebel Server and client:
   a. On the Security tab, select the appropriate user or group name and make sure that Full Control is selected on the Permissions list.
   b. Click Advanced.
   c. On the Access Control Settings dialog box, make sure that you check the option to allow inheritable permissions from the parent to propagate to this object.
8. To close the Permissions and File Properties dialog boxes, click OK.

Creating the Siebel Service Owner Account

This task is part of *Planning Your Siebel Deployment*.

(Windows) The Siebel Enterprise Server requires that you create a standard Windows system user account. This account must be available on each Siebel Server in your Siebel Enterprise under which Siebel services and components operate.

(UNIX) The Siebel Enterprise Server requires that you create a standard UNIX system user account. This account must be available on each Siebel Server in your Siebel Enterprise under which Siebel processes and components operate. (For convenience, the term Siebel service owner account is used here for both Windows and UNIX.)

Use the following guidelines to create the Siebel service owner account:

- The Siebel service owner account must be defined or available on each applicable server computer: on the Siebel Gateway, on each Siebel Server in the Siebel Enterprise, and on any computer on which the Siebel File System exists.
- (Windows) The Siebel service owner account must be part of a Windows domain, so that services are operated under the same account on all of the Windows servers.

  If you are using a local account instead, then you must set up that account to be identical on each server, using the same login ID and password.
- (Windows) The Siebel service owner account must be part of the administrator group. The Siebel service owner account might be the same administrator account under which the Siebel CRM modules are installed, or a different account that is part of the administrator group.
- (Windows) The Siebel service owner account must have the following Windows rights and privileges:
  - Log on as a Service
  - Act as part of the operating system

  > **Note:** Group policy must also permit these rights for local policy.
- (UNIX) Create the Siebel service owner account at the network level, using an appropriate administration tool for your UNIX operating system, so that the same account can be used for all of the UNIX server computers within the Siebel Enterprise Server. Make sure that the numeric values for uid and gid match across the various computers.
- The Siebel service owner account password must not require a change on next logon and must be set not to expire.
- The Siebel service owner account name or password cannot contain any spaces.
- The Siebel service owner account must have read-write access to the Siebel File System and the Migration Package Location. Specifically, this account must have access to the primary Siebel File System specified in the Enterprise Profile you deploy. (The same is true of the Siebel Application Interface owner account.)
- The Siebel service owner account name must not exceed 30 characters in length, including domain information.
- When you determine the Siebel service owner account name and password, record this information in your copy of the worksheet in *Siebel Deployment Planning Worksheet*. (For security reasons, you might prefer not to record the password in the worksheet.) See also *Siebel Security Guide* .

Related Topics

About the Migration Package Location

(UNIX) *Setting Permissions and Ownership on UNIX*

Siebel File System Requirements

Siebel Deployment Planning Worksheet

Requirements for Installing and Configuring Siebel Enterprise Components

Review the requirements and guidelines in the following topics before installing and configuring the Siebel Enterprise Components software:

- *General Requirements for Installing and Configuring Siebel Enterprise Components*
- *Requirements for Siebel Gateway Authentication*
- *Requirements for Installing and Configuring the Siebel Gateway*
- *Requirements for Installing and Configuring the Siebel Server*
- *Requirements for Installing the Siebel EAI Connector*

For requirements specific to the Siebel database, see *Requirements for Installing and Configuring the Siebel Database*.

General Requirements for Installing and Configuring Siebel Enterprise Components

This topic is part of *Requirements for Installing and Configuring Siebel Enterprise Components*.

Review this topic before installing and configuring Siebel CRM server software. See also the topics about individual modules.

- Before installing Siebel CRM software, check for applicable alerts, bulletins, or other documents on My Oracle Support.
- Review *About Installing Siebel CRM* and *Planning Your Siebel Deployment*.
- All computers or operating system instances on which the Siebel Enterprise Components software is installed must meet the hardware and software requirements. The Siebel CRM installer verifies that you have required software at the necessary version levels.
- You must have created a Siebel CRM installation image that includes all of the modules and components that you require. Users who run the Siebel CRM installer or modify installer files must have the necessary permissions to the network directories where the Siebel image is located.
- Where you install Siebel Enterprise Components software on the same computer, then all modules are installed into a common root directory, as follows:
    - (Windows) For example, the installation directory might be `c:\siebel`. For more information, see *About Installing Siebel CRM* and *Installation Tasks for Siebel CRM*.

  - (UNIX) For example, the installation directory might be `/Siebel`. The environment variable SIEBEL_ROOT might be set differently in different locations, to reflect the full path to the installation location of a specific component, such as Siebel Server. For more information, see *About Installing Siebel CRM* and *Installation Tasks for Siebel CRM*.

- Installation directories must meet the requirements described in *File and Directory Naming Conventions*.
- If you will be clustering the Siebel Gateway or Siebel Server, then plan your use of clustering or redundant disk arrays (RAID) to configure against a single point of failure. For information about the recommended native clustering feature for Siebel Gateway, see *Configuring the Siebel Gateway Cluster*. See also *Siebel Deployment Planning Guide* .
- Each computer or operating system instance that supports Siebel CRM server software must have TCP/IP network connectivity to other computers that are part of or that work with the Siebel Enterprise Server. For example, the Siebel Gateway computer requires connectivity to all of the Siebel Server computers. Verify connectivity between all such computers, using the `ping` utility. For more information, see *Verifying Network Connectivity for the Siebel Server Computer*.
- Before running any of the Siebel Management Console tasks in which you will specify a server computer's host name or IP address, validate the host name or IP address that you will enter during configuration. You can use the `ping` utility for this purpose. The host name and IP address data that you specify when you run the Siebel Management Console must be correct in order for the configuration process to proceed correctly.
- Verify that the network names of the servers that will support the Siebel Gateway and all of the Siebel Servers are recorded in *Siebel Deployment Planning Worksheet*. You need this information when configuring the Siebel Servers.
- It is strongly recommended to review all applicable security information before you configure Siebel CRM, which uses Transport Layer Security (TLS).
- When data encryption is enabled, make a backup copy of the key file before you start a migration installation for Siebel Enterprise Server. Copy back this file after the migration.

  > **CAUTION:** When data encryption is enabled, a migration installation for Siebel Enterprise Server creates a new key file (keyfile.bin), overwriting your existing key file. If this happens, the encrypted columns will be inaccessible. To prevent this result, you must make a backup copy of the key file before starting the migration installation. After the migration installation, copy back the original key file. For more information about data encryption, see *Siebel Security Guide* .

- Review the issues described in *Managing Temporary Disk Space Required by Siebel Installers and Wizards*. For example, make sure that you have adequate disk space.
- It is strongly recommended that you install and deploy, or at least install, all of the languages that you expect to require. For more information, see *Installing and Deploying Siebel CRM with Multiple Languages*.

  For more information, see additional relevant topics, including:

  - *About Installing Siebel CRM*
  - *Planning RDBMS Installation and Configuration*
  - *About Configuring Siebel CRM* and following topics
  - *Preparing to Run Siebel Server Components*

- You cannot install new Siebel Enterprise Server modules into an existing installation.
- Note that the Siebel CRM installer user interface can run in the supported languages described in *About the Language in Which Siebel Installers and Wizards Run*.

- Before you configure Siebel Enterprise Server modules, you must have created the Siebel File System. It must meet all of the criteria described in *Creating the Siebel File System*.
- Before you install Siebel CRM, review documented information about the installation and configuration process and applicable requirements. Customers must manage the overall installation and configuration process carefully to ensure success.
- For existing deployments, all Siebel Server component jobs must complete before:
  - You perform any Siebel CRM installations in a migration installation case.
  - You perform any upgrade tasks in a full database upgrade case.

  After you perform all installation and upgrade tasks, recreate any component job definitions that you require for your deployment.
- Administrative rights are required for installation or uninstallation of Siebel CRM software. For information about setting administration rights, consult the operating system manuals applicable to your installations.
- Antivirus software, firewalls, and other security software can interfere with installation and operation of software if they are not properly configured. In some cases, you might need to disable antivirus software while installing Siebel CRM software. It is also recommended to exclude certain types of files from virus scanning. In addition, you might need to disable the Windows feature User Account Control while installing Siebel CRM software. Follow these guidelines for using antivirus software:
  - Run virus scans on Siebel Enterprise Servers during maintenance windows, if possible, not when they are online. Scans can then be performed on all files at that time.
  - You can run regular virus scans for the Siebel File System. It is recommended to perform these scans during off-peak hours.
- For a migration installation case, you must perform the migration installation for Siebel Application Interface before you perform the migration installations for any other Siebel CRM server modules. When you do the migration installations for other Siebel CRM modules, the installer prompts you to provide the host name and installation location for the migrated installation of Siebel Application Interface. Without this information, the migration installations of Siebel Gateway, Siebel Server, or other modules cannot complete successfully.
- Additional requirements apply for migration installations of the current release. For more information, see *Additional Tasks for Migration Installations*.

Related Topics

Overview of Installing Siebel CRM

About Installing Siebel CRM

Installation Tasks for Siebel CRM

Additional Tasks for Migration Installations

About Configuring Siebel CRM and following topics

Related Books

Siebel Security Guide

Siebel Deployment Planning Guide

Requirements for Siebel Gateway Authentication

This topic is part of *Requirements for Installing and Configuring Siebel Enterprise Components*.

Before you configure any Siebel Enterprise Components software, review the requirements associated with authentication for access to the Siebel Gateway. Authentication applies to access using Siebel Management Console or Siebel Server Manager.

Note the following requirements:

- Supported Siebel Gateway authentication methods include database authentication, LDAP, and custom methods. Additional configuration is required. You can use the same authentication method as for your Siebel enterprise connections, or use different methods.
- When you initially configure the Siebel Enterprise (after installing and configuring the Siebel Gateway), you propagate settings to the Siebel Gateway that specify authentication settings. Siebel Gateway authentication uses Siebel security adapters, which are discussed in detail in *Siebel Security Guide* .
- The user account that you use for Siebel Gateway authentication must have the same privileges as the Siebel administrator account created during the Siebel installation process; these privileges are required to connect to the Siebel Gateway. You can choose to use the Siebel administrator account for Siebel Gateway authentication, or you can create a new database user account, ensuring that you assign it the same level of rights and privileges as the Siebel administrator account. For more information about requirements for this account, see *Siebel Security Guide* .
- If you do not have an existing Siebel database, then, if you will use database authentication, you must install it before you can configure the Siebel CRM software using the Siebel Management Console.
- After you configure the Siebel Gateway, users doing subsequent Siebel Management Console tasks that require Siebel Gateway authentication, or using Siebel Server Manager, must enter Siebel administrator user credentials that are valid for the authentication method in use.

  All Siebel Management Console tasks are subject to Siebel Gateway authentication.

  For more information about using Server Manager, see *Siebel System Administration Guide* .
- If you want to use LDAP or a custom authentication method for the Siebel Gateway, then you can configure this method as part of the Siebel Management Console tasks for creating or modifying the Siebel Enterprise. The Siebel Management Console both configures and enables the security adapters (for the Siebel Enterprise or for the Siebel Gateway).
- When you configure the Siebel Gateway using Siebel Management Console, the Authorization Roles setting defines the role that is required to access the Siebel Gateway. This setting includes the Siebel Administrator, which is provided by Oracle and cannot be modified. Optionally, you can add more comma-separated roles, specifying other roles that can access the Siebel Gateway.
- If you want to configure a second Siebel Enterprise on the same Siebel Gateway, then do not check the option to propagate authentication settings to the Siebel Gateway when prompted. Otherwise, your settings will overwrite the existing settings.

Related Topics

Configuring the Siebel Gateway and Security

Configuring the Siebel Enterprise

Requirements for Installing and Configuring the Siebel Gateway

This topic is part of *Requirements for Installing and Configuring Siebel Enterprise Components*.

Review this topic before installing and configuring the Siebel Gateway. See also *General Requirements for Installing and Configuring Siebel Enterprise Components*.

- Install the Siebel Gateway at least once for each Siebel Enterprise. If necessary, multiple Siebel Enterprises can be supported by a single Siebel Gateway.

  - (Windows) Installing multiple instances of the same version of Siebel Gateway on the same computer is not supported.
  - (UNIX) In general, installing multiple instances of the same version of Siebel Gateway on the same computer is not recommended.

  For more information, see *Planning to Install Multiple Instances of Siebel CRM*. This document was previously published as Siebel Technical Note 531.
- Optionally install multiple instances of Siebel Gateway on different nodes in order to configure and deploy clustering. For more information, see *Configuring the Siebel Gateway Cluster*.
- The default port number for Siebel Gateway is 2320. If necessary, you can select any port number (32767 or lower) that is free on the computer where the Siebel Gateway is running. Do not use port number 2321, which is the default port number for the SCBroker (Siebel Connection Broker) component, or any other port that is already in use on the server.

  (UNIX) To confirm that a port (such as 2320) is free, use a command like `netstat -a | grep 2320`.

Related Topics

Configuring the Siebel Gateway and Security

Configuring the Siebel Gateway Cluster

Requirements for Installing and Configuring the Siebel Server

This topic is part of *Requirements for Installing and Configuring Siebel Enterprise Components*.

Review this topic before installing and configuring the Siebel Server. See also *General Requirements for Installing and Configuring Siebel Enterprise Components*.

- To configure a Siebel Server, a Siebel Gateway must be installed and running, and the Siebel Enterprise must be configured.
- After you install the Siebel Gateway, you run the Siebel Management Console to configure it, and then you configure the Siebel Enterprise. All Siebel Servers you install that are part of the same Siebel Enterprise, regardless of the operating system, must connect to the same Siebel database. For most deployments, all of the Siebel Servers connecting to this database will belong to the same Siebel Enterprise.

  Additional Siebel Servers that you install and configure inherit parameters from the Siebel Enterprise. You configure each Siebel Server using the Siebel Management Console.

- When you run the Siebel Management Console to configure each Siebel Server, all of the component groups are listed and you must enable the ones that you need for this server. If you do not enable component groups during Siebel Server configuration, then you can enable them manually after installation, using Siebel Server Manager. For more information about component groups and about using Server Manager, see *Siebel System Administration Guide* . See also *Preparing to Run Siebel Server Components*.
- Depending on your business requirements, you might deploy one or more Siebel Enterprise Servers. For information about deploying multiple Siebel Enterprise Servers, see *Siebel Deployment Planning Guide* . This document was previously published as Siebel Technical Note 531.

> **Note:** In special cases, such as for some large deployments, a single Siebel database can support multiple Siebel Enterprises. Such a deployment must be planned carefully. For details, see 477829.1 (Article ID) on My Oracle Support. This document was previously published as Siebel Technical Note 544.

This topic includes the following information:

- *About Clustering the Siebel Gateway and Siebel Server*
- *Language and Locale Requirements*
- *Guidelines for Installing Multiple Languages on the Siebel Server*
- *Search Server*
- *Database Requirements*
- *Third-Party Software Requirements*
- *Temporary Disk Space*

About Clustering the Siebel Gateway and Siebel Server

This topic is part of *Requirements for Installing and Configuring the Siebel Server*.

If you will operate certain servers as part of a cluster, then it is strongly recommended that you install and configure the Siebel Gateway and the Siebel Server on separate resource groups. For information about clustering, see *Siebel Deployment Planning Guide* .

For information about configuring Siebel Gateway clustering using Siebel Management Console, see *Configuring the Siebel Gateway Cluster*.

Language and Locale Requirements

This topic is part of *Requirements for Installing and Configuring the Siebel Server*.

Make sure that you have the appropriate locales installed on the computers on which you intend to deploy Siebel Server. Siebel log and configuration files use UTF-8 with Byte Order Mark as the default encoding. If these files contain any non-ASCII characters, then proper viewing and editing requires a UTF-8 locale.

The locale can affect how dates and times are displayed. For more information, see *Siebel Global Deployment Guide* .

> **Note:** (UNIX) Make sure that the locale that you are using matches the locale set at your operating system level. For details, see *Specifying the Locale for Your UNIX Operating System*.

Platform, database, and other certifications for Siebel are found on My Oracle Support. For help searching for certifications, see *How to Check the Certification Matrix in the New My Oracle Support Site* (KB871388).

Guidelines for Installing Multiple Languages on the Siebel Server

This topic is part of *Requirements for Installing and Configuring the Siebel Server*.

If you will install multiple language versions of Siebel applications on your Siebel Servers, then review the following configuration guidelines:

- You can include multiple languages when you install and configure a Siebel Server.
- For each installed language that you choose to deploy when you configure the Siebel Server, language-specific Application Object Manager components are created.

  For more information about scenarios for multilingual deployments, see *Installing and Deploying Siebel CRM with Multiple Languages* and related topics. See also *Siebel Global Deployment Guide* .
- If multiple languages are installed on a Siebel Gateway or Siebel Server, then you will be prompted for the primary (base) language. The primary (base) language is the language in which you want your server to run and in which you normally want to read messages. If you want to change the language in which you bring up your server, then you must change this setting.

Search Server

This topic is part of *Requirements for Installing and Configuring the Siebel Server*.

A Siebel Server computer can be configured to execute searching using a locally installed search server or can be pointed to a remote search server to handle search execution tasks.

Note: For search-related installation guidelines and requirements, see *Siebel Search Administration Guide* .

Database Requirements

This topic is part of *Requirements for Installing and Configuring the Siebel Server*.

Make sure that your database administrator has installed the RDBMS that your site will use and created the Siebel database instance and that you meet all of the database connectivity requirements. For more information, see *Configuring the RDBMS*, including *Configuring Siebel Server Connectivity to the Siebel Database*. See also *Planning RDBMS Installation and Configuration*.

The RDBMS that you are using must support the same languages and code pages that you install on the Siebel Servers.

Platform, database, and other certifications for Siebel are found on My Oracle Support. For help searching for certifications, see *How to Check the Certification Matrix in the New My Oracle Support Site* (KB871388).

Note: After performing a migration installation of the current release, then you must run Incremental Repository Manager. For more information, see *About Database Updates for Siebel CRM*.

Third-Party Software Requirements

This topic is part of *Requirements for Installing and Configuring the Siebel Server*.

Make sure that you have already installed the appropriate version of all of the third-party software products required. Otherwise, the Required Software Components prompt appears.

Platform, database, and other certifications for Siebel are found on My Oracle Support. For help searching for certifications, see *How to Check the Certification Matrix in the New My Oracle Support Site* (KB871388).

Temporary Disk Space

This topic is part of *Requirements for Installing and Configuring the Siebel Server*.

Review the issues described in *Managing Temporary Disk Space Required by Siebel Installers and Wizards*. For example, make sure that you have adequate disk space.

Requirements for Installing the Siebel EAI Connector

This topic is part of *Requirements for Installing and Configuring Siebel Enterprise Components*.

EAI Connector is included when you install Siebel Enterprise Components. You can use it alongside other installed components or install Siebel Enterprise Components in a separate location specifically for using EAI Connector.

See also: *General Requirements for Installing and Configuring Siebel Enterprise Components*.

Platform, database, and other certifications for Siebel are found on My Oracle Support. For help searching for certifications, see *How to Check the Certification Matrix in the New My Oracle Support Site* (KB871388).

Requirements for Installing and Configuring the Siebel Application Interface

Before installing and configuring the Siebel Application Interface, review the requirements described in the following information:

- *About the Siebel Application Interface*
- *Planning the Siebel Application Interface Topology*
- *General Requirements for Installing and Configuring the Siebel Application Interface*
- *Requirements for User Permissions for the Siebel Application Interface*
- *Requirements for Configuring Anonymous Users for the Siebel Application Interface*

About the Siebel Application Interface

This topic is part of *Requirements for Installing and Configuring the Siebel Application Interface*.

Siebel Application Interface enables communication between Siebel Web Clients and Siebel Servers. Siebel applications are a family of Web-based applications that users access through a standard Web browser. Several components work together to deliver the applications to end users:

- **Siebel Web Client.** The Siebel CRM client that runs in the browser on the end user's computer and connects to the Siebel Application Interface.

  For information about other Siebel clients, see the applicable documentation. For example, for information about the Siebel Mobile applications, see *Siebel Mobile Guide: Connected* and *Siebel Mobile Guide: Disconnected* .

- **Siebel Application Interface.** A program that communicates with the Siebel Web Engine (which is part of an Application Object Manager component, such as Call Center Object Manager) on the Siebel Server and with browsers run by users of Siebel CRM.

  Siebel Application Interface is installed with Siebel Enterprise Components, as described in *Installing Siebel CRM Using the User Interface*. For configuration instructions, see *Configuring the Siebel Application Interface*. See also *Planning the Siebel Application Interface Topology*.
- **Siebel Web Engine (part of Application Object Manager on Siebel Server).** The Siebel Web Engine is part of the Application Object Manager component on the Siebel Server, such as Call Center Object Manager. The Application Object Manager provides access to Siebel CRM data and logic.

Planning the Siebel Application Interface Topology

This task is part of *Requirements for Installing and Configuring the Siebel Application Interface*.

Before you install the Siebel Application Interface for your Siebel deployment, as described in *Siebel CRM Download and Installation*, you must decide how you will distribute instances of Siebel Application Interface and other Siebel CRM modules. For the best performance and scalability, install the Siebel Application Interface onto one or more dedicated computers.

Two basic methods are as follows:

- **Single-node.** Siebel Application Interface is installed with Siebel Enterprise Components. You might choose to install Siebel Enterprise Components once and configure and use Siebel Application Interface on a single computer node, or operating system instance. Install Siebel Application Interface in the secured application intranet zone, in a different location on the same computer as the other Siebel CRM modules, or on a different computer.

  You can use existing Web servers to host reverse proxy or authentication, where these products support jsessionid cookies and the mod_jk module. All instances of Siebel Application Interface must be protected by a reverse proxy.
- **Distributed.** Distributing Siebel CRM modules, where instances of Siebel Application Interface on one or more dedicated computers or operating system instances connect to multiple Siebel Servers in the Siebel Enterprise. These Siebel Application Interface instances can be dynamically balanced for Application Object Manager components on different Siebel Server computers.

  You can distribute incoming network traffic over multiple Siebel Application Interface instances by using third-party HTTP load balancers that support jsession-based load balancing in front of a reverse proxy Web server. You can use existing Web servers to host reverse proxy, authentication, or load balancing, where these products

support jsessionid cookies and the mod_jk module. All instances of Siebel Application Interface must be protected by a reverse proxy.

> **Note:** Application containers for Siebel Application Interface instances on multiple nodes can be load balanced using products such as Apache HTTP Server (httpd) with the mod_jk module. The products you use must provide reverse proxy services and support load balancing based on jsessionid cookies. All instances of Siebel Application Interface must be protected by a reverse proxy. For more information about these or other products that meet these requirements, refer to vendor documentation. If you configure load balancing for application containers in this way, then, whenever an application container fails for a Siebel Application Interface node, the active sessions handled by that node transparently failover to another available healthy application container node. For more information, see *Load Balancing Siebel Application Interface by Configuring the workers.properties File*.

Each deployment choice involves a trade-off. However, in enterprise-sized deployments, it is strongly recommended that you use a distributed node deployment for the following reasons:

- **Less resource contention.** Distributing the Siebel Application Interfaces and the Siebel Servers (with Application Object Manager components) on different computers eliminates contention for CPU and other server resources. However, to take advantage of the performance improvement, you must have a high-speed network connection between the two computers.
- **Higher fault tolerance.** Operating multiple instances of components on multiple computers reduces downtime and the effect of failure on any one computer.
- **High availability.** A multinode configuration is required for deployments that support large numbers of concurrent users or where high availability is an operating requirement.

> **Note:** Configuring a maintenance window page in Siebel CRM can be performed in multiple ways:
> - If a reverse proxy is used, the reverse proxy can be set to use error documents. An example for Apache HTTP follows: `ErrorDocument 503 /maintenance-message.html`
> - URL rewrite capability of the web server can also be used for this.

Related Topics

About the Siebel Application Interface

Installing Siebel CRM in a Migration Installation

Configuring the Siebel Application Interface

Customizing the Application Container for Siebel Application Interface

Installing Multiple Instances of the Same Version of Siebel CRM

Load Balancing Siebel Application Interface by Configuring the workers.properties File

Related Books

Siebel Deployment Planning Guide

Siebel Security Guide

General Requirements for Installing and Configuring the Siebel Application Interface

This topic is part of *Requirements for Installing and Configuring the Siebel Application Interface*.

This topic lists general requirements for installing and configuring the Siebel Application Interface.

- Before installing Siebel CRM software, check for applicable alerts, bulletins, or other documents on My Oracle Support.
- Review the issues described in *Managing Temporary Disk Space Required by Siebel Installers and Wizards*. For example, make sure that you have adequate disk space, and make sure that the login ID running the installer has write permissions to the temporary directory.
- Make sure that the computers on which you install the Siebel Application Interface meet all of the hardware and software platform requirements.

  Platform, database, and other certifications for Siebel are found on My Oracle Support. For help searching for certifications, see *How to Check the Certification Matrix in the New My Oracle Support Site* (KB871388).
- You install the Siebel Application Interface as described in *Installing Siebel CRM Using the User Interface*.
- Before you can configure Siebel Application Interface using Siebel Management Console, as described in *Configuring the Siebel Application Interface*, you must have configured the Siebel Gateway, the Siebel Enterprise, and the Siebel Server.

  Application Object Manager components must be enabled for the Siebel CRM products that you purchased and intend to use. You enable components during initial Siebel Server configuration. For information about enabling server components using Siebel Server Manager, see *Siebel System Administration Guide* .

  > **Note:** Wait to install additional Siebel Servers until after you have completed installation of the Siebel Application Interface software and verified the connection from the Siebel Application Interface to the initial Siebel Server.

- You must perform the migration installation for Siebel Application Interface before you perform the migration installations for any other Siebel CRM server modules. When you do the migration installations for other Siebel Enterprise Server modules, the installer prompts you to provide the host name and installation location for the migrated installation of Siebel Application Interface. Without this information, the migration installations of Siebel Gateway, Siebel Server, or other modules cannot complete successfully.
- You can deploy multiple languages on one Siebel Application Interface instance. The Siebel Server and the Siebel Application Interface do not have to be operated in the same language. However, the Siebel Server, the Siebel Application Interface, and all of the other server components must use the same character set. For more information, see *Siebel Global Deployment Guide* . See also *Installing and Deploying Siebel CRM with Multiple Languages*.
- At least one Siebel Application Interface is required for each Siebel Enterprise.
- Note that uninstalling a Siebel Application Interface instance that you have configured removes the associated Siebel application configuration. See also *Uninstalling Siebel CRM*.
- Note that the user interface of the Siebel CRM installer, which you use to install Siebel Application Interface, can run in the supported languages described in *About the Language in Which Siebel Installers and Wizards Run*.
- As noted in *Installing Siebel CRM in an Update Installation*, if you are updating an installed release of Siebel CRM 17.x or 18.x (through 18.4) to the current release, then you must update all installed instances of Siebel

Application Interface before you update other installed Siebel CRM modules. If you are updating Siebel CRM 18.5 or later to the current release, then you can update the installed modules in any sequence.

Requirements for User Permissions for the Siebel Application Interface

This topic is part of *Requirements for Installing and Configuring the Siebel Application Interface*.

This topic describes some requirements for permissions that affect installation or operation of the Siebel Application Interface and the Siebel application configurations that are associated with it.

The user who will run the Siebel Application Interface must have read, write, and execute permissions on the `SIEBEL_AI_ROOT\applicationcontainer_external\siebelwebroot` directory and subdirectories (in this path, `SIEBEL_AI_ROOT` is the directory in which the Siebel Application Interface is installed). These permissions allow static public files to be cached on the Siebel Application Interface. These directories are created during Siebel Application Interface installation and configuration. In general, Siebel administrators require full access to `SIEBEL_AI_ROOT` and all of its subdirectories.

Requirements for Configuring Anonymous Users for the Siebel Application Interface

This topic is part of *Requirements for Installing and Configuring the Siebel Application Interface*.

As noted in *Creating a Siebel Application Interface Profile*, when you configure profiles for Siebel CRM using Siebel Management Console, you are prompted for user names and passwords of Siebel users who will serve as anonymous users for Siebel applications. Provide the anonymous user credentials, as appropriate for your deployment.

The anonymous user has functions that include the following:

- Starting the anonymous session that displays the login page to an end user for an application. This function applies to both employee applications and customer applications.
- Allowing anonymous browsing and navigation within the application. In this case, usually applicable to customer applications only, the anonymous user must have access to any public view in the application.

You can define as many different anonymous users as you require for your deployment, each with the appropriate permissions, according to your needs. During Siebel software configuration, using Siebel Management Console, you can specify an anonymous user name and associated password when you:

- Configure the Siebel Server profile, as described in *Configuring the Siebel Server*. You can specify the anonymous user to serve as a default for that Siebel Server.
- Configure the Siebel Application Interface profile, as described in *Configuring the Siebel Application Interface*:
    - An anonymous user that will serve as a global default for all applications, unless overridden by the corresponding Siebel Server profile setting or by the corresponding setting for an individual application.
    - An anonymous user that will override the default for an individual application, unless overridden by the corresponding Siebel Server profile setting.
    - An anonymous user that will be used for REST inbound authentication. The password for this anonymous user is used for REST requests that are sent without a user name and password.

If, after performing the initial configuration, you change the password for the database account that you are using for an anonymous user, or decide to specify a different anonymous user for a particular application, then you can modify the configuration to provide the updated password or to specify a different anonymous user for this application.

Record the credentials for each anonymous user in your deployment in the worksheet provided in *Siebel Deployment Planning Worksheet*.

Note the following guidelines and requirements:

- For more information about configuring anonymous users, see *Siebel Security Guide* .
- All anonymous users for your deployment must be defined in the Siebel database and must be defined as Siebel users. Strong passwords are highly recommended for anonymous user accounts. Anonymous users should have limited access rights, no more than is necessary for the usage context. The Siebel Management Console does not allow the Siebel administrator user, SADMIN, to be specified as an anonymous user.
- You can create new database users that will serve as anonymous users through editing the grantusr.sql script. You review, modify, and execute this script before you install the Siebel database and configure the Siebel deployment. GUESTCST is an example of an anonymous user account that you would create in the grantusr.sql script. After you install the Siebel database and configure the Siebel deployment, then you still also add corresponding Siebel users. For more information, see *Creating Table Owner and Administrator Accounts*.
- The anonymous user for employee applications must be defined as an employee and must be associated with an appropriate position and responsibility.
- If your Siebel application does not use functionality that requires anonymous browsing, then you can disable access by the anonymous user by setting the AllowAnonUsers parameter for the Application Object Manager component to False. If anonymous browsing must be enabled, then you set AllowAnonUsers to True.
- Anonymous users are used by applications that allow anonymous browsing by unregistered users, that display a Siebel login page, that support self-registration, or that support external authentication such as LDAP. The anonymous user and password are used for serving the login page.
- To use nondefault or application-specific anonymous users, you must specify the anonymous user and passwords for each application, in the Siebel Application Interface profile. You can also specify the anonymous user in the Siebel Server profile.

Related Topics

Creating a Siebel Application Interface Profile

Creating Table Owner and Administrator Accounts

Related Books

Siebel Security Guide

Siebel System Administration Guide

Deploying Siebel Open UI

Requirements for Installing and Configuring the Siebel Database

Before you install and configure the Siebel database, review the following information:

- For Siebel CRM language support, Unicode support, and legacy code page support, see the Global Deployment Guide.

  Platform, database, and other certifications for Siebel are found on My Oracle Support. For help searching for certifications, see *How to Check the Certification Matrix in the New My Oracle Support Site* (KB871388).
- For new installations, make sure that the Siebel database instance has been created and is properly configured, as documented in *Configuring the RDBMS*.
- Review *Installing the Siebel Database on the RDBMS*, including *About Installing the Siebel Database*, and the topics that follow.
- Obtain the services of a qualified database administrator (DBA) to assist you with your installation and, where applicable, upgrade.
- Complete the appropriate RDBMS-specific information in your copy of the worksheet in *Siebel Deployment Planning Worksheet*, as follows.

This topic includes the following information:

- *General Requirement*
- *Oracle Database*
- *IBM DB2*
- *Microsoft SQL Server/Azure SQL Database Servers*

General Requirement

This topic is part of *Requirements for Installing and Configuring the Siebel Database*.

Installing the Siebel database is subject to installation requirements for Siebel Database Configuration Utilities related to Siebel Gateway authentication, which are noted in *General Requirements for Installing and Configuring Siebel Enterprise Components*.

Oracle Database

This topic is part of *Requirements for Installing and Configuring the Siebel Database*.

For Oracle Database, make sure that the following elements are defined for your Siebel database:

- **Oracle SQLNet alias connect string.** Provide the name of the Oracle schema qualifier or table owner name and the SQLNet connect string for Oracle. You need this to connect to your Oracle Database.
- **Table owner account (schema) name and password.** Using an Oracle Database requires that you assign a user name and password to any database tables that you create. The term *table owner* refers to the schema that owns the database objects, such as tables, indexes, views, and triggers.

Before installing the Siebel database components, you have to edit the grantusr.sql script, enter this and related information, and execute the script. See *Creating Table Owner and Administrator Accounts*.

- **Siebel data tablespace.** The name of the tablespace on the Oracle Database server where the Siebel data tables are stored.
- **Siebel index tablespace.** The name of the tablespace on the Oracle Database server where the Siebel indexes are stored.
- Use the Oracle Database Client software on a client or server installed with Siebel. Oracle Database Client is provided as a part of with the Siebel CRM media.
- Siebel installer will install the Oracle database client on all platforms by default under a new folder named "Oracledbclient" under <SIEBEL_HOME>/oracledbclient folder.

Platform, database, and other certifications for Siebel are found on My Oracle Support. For help searching for certifications, see *How to Check the Certification Matrix in the New My Oracle Support Site* (KB871388).

> **Note:** Make sure that the Oracle Database Client and Oracle Database versions are compatible, unless you are otherwise advised by Oracle Support or unless required for your particular deployment.

IBM DB2

This topic is part of *Requirements for Installing and Configuring the Siebel Database*.

For IBM DB2, make sure that the following elements are defined for your Siebel database:

- **Database alias.** The DB2 database alias that you created when you installed the DB2 software.
- **Table owner or database owner account user name and password.** DB2 requires that you assign a user name and password to each database that you create.

  Before you install the tablespaces and indexes, you are prompted to edit the grantusr.sql script, enter this and related information, and execute the script. See *Creating Table Owner and Administrator Accounts*.
- **Siebel index tablespace.** The name of the tablespace on the DB2 server where the Siebel indexes are stored.
- **Siebel 4-KB tablespace.** The name of the tablespace on the DB2 server where the 4 KB Siebel data tables are stored.
- **Siebel 16-KB tablespace.** The name of the tablespace on the DB2 server where tables reside whose row length is equal to or greater than 4005 bytes, but less than 16384 bytes.
- **Siebel 32-KB tablespace.** The name of the tablespace on the DB2 server where tables reside whose row length is 32768 bytes.
- Statistics are generated automatically during table, index, and seed data installation, and during the Siebel Repository import process. However, it is recommended that statistics be kept up to date through standard database administration procedures.
- Make sure that you have installed the required IBM updates on your database server computer.
- Make sure that IBM DB2 is properly configured and you have allocated disk space appropriate to your installation requirements.

Microsoft SQL Server

This topic is part of *Requirements for Installing and Configuring the Siebel Database*.

Make sure that the following elements are defined for your Siebel database:

- Database name
- Database owner account username and password

  Before installing the Siebel database components, you have to edit the grantusr*.sql scripts, enter this and related information, and execute the script. See *Creating Table Owner and Administrator Accounts*.

Requirements for Installing Siebel Web Clients

Review the issues and tasks in this topic before installing the Siebel Web Client, as described in *About Installing Siebel Web Client or Siebel Tools*.

This topic includes the following information:

- *Administrative Rights for Installation*
- *Directory Naming Conventions*
- *Requirements and Recommendations for Siebel Web Client*
- *Restricted Support for Siebel Developer Web Client*
- *Requirements for Oracle Database SE2 and the Local Database*
- *Security-Related Software*
- *Other Oracle or Third-Party Software*
- *Chrome Browser Users*
- *Closing Browser Sessions*
- *Migration Installations*

Administrative Rights for Installation

This topic is part of *Requirements for Installing Siebel Web Clients*.

Administrative rights are required for installation or uninstallation of the Siebel Web Client.

For information about setting administration rights, consult the operating system manuals for the version of Microsoft Windows on which you are installing the Siebel Web Client software.

For information about uninstalling Siebel Web Client software, see *Uninstalling Siebel CRM*.

Directory Naming Conventions

This topic is part of *Requirements for Installing Siebel Web Clients*.

By default, the Siebel Web Client installer assumes an installation directory of `c:\Siebel\Client` or a similar location. You can specify to install in a different location.

If you must change the installation location to use a nondefault path, then use installation directory names that describe the Siebel CRM module being installed. The directory name can use any characters appropriate for Windows-compatible long file names, except that spaces must not be used in your install directory.

In this guide, the directory into which you install the Siebel Web Client is often referred to as `SIEBEL_CLIENT_ROOT` .

> **CAUTION:** Do not install other Siebel CRM modules, such as Siebel Tools, into the same directory where you have installed the Siebel Web Client, such as `c:\Siebel\Client`. Install each module into a separate directory. If you install multiple Siebel CRM modules on the same computer (that will access different data sources), then determine your directory-naming convention before you begin installing.

Requirements and Recommendations for Siebel Web Client

This topic is part of *Requirements for Installing Siebel Web Clients*.

Before beginning Siebel Web Client installation, review the requirements and recommendations for hardware, system software, and third-party software.

Restricted Support for Siebel Developer Web Client

This topic is part of *Requirements for Installing Siebel Web Clients*.

The Siebel Developer Web Client is not supported for end-user deployment. This Siebel Web Client type is supported only for development, troubleshooting, and limited administration usage scenarios.

Requirements for Oracle Database SE2 and the Local Database

This topic is part of *Requirements for Installing Siebel Web Clients*.

In the current release, Oracle Database SE2 is used for the local database, and the sample database is no longer supported. For information about the requirements for installing and using Oracle Database SE2 for Siebel Mobile Web Client, see *Installing and Using Oracle Database SE2 for the Local Database*.

Security-Related Software

This topic is part of *Requirements for Installing Siebel Web Clients*.

Antivirus software, firewalls, and other security software can interfere with installation and operation of software if they are not properly configured. In some cases, you might need to disable antivirus software while installing Siebel CRM software, including Oracle Database SE2 for the local database for Siebel Mobile Web Client on Microsoft Windows. It is also recommended to exclude certain types of files from virus scanning. In addition, you might need to disable the Windows feature User Account Control while installing Siebel CRM software.

Other Oracle or Third-Party Software

This topic is part of *Requirements for Installing Siebel Web Clients*.

Your Siebel application might require some other Oracle software products or third-party software products to be installed on the client computer, for full functionality.

> **Note:** Remember to install on the client computers all of the software required to view any standard attachment types your Siebel CRM deployment might use.

For information about requirements for the web browser environment for running Siebel Web Clients, see *Deploying Siebel Open UI* .

Also see these topics:

- *About Installing Siebel Web Client or Siebel Tools*
- *Installing and Using Oracle Database SE2 for the Local Database*
- *Installing Desktop Integration Siebel Agent on Client Computers*
- *Siebel Client Deployment Requirements When Using Firewalls or Proxy Servers*
- *Configuring Siebel CRM for Pop-Up Blocker Compatibility*

Platform, database, and other certifications for Siebel are found on My Oracle Support. For help searching for certifications, see *How to Check the Certification Matrix in the New My Oracle Support Site* (KB871388).

Chrome Browser Users

This topic is part of *Requirements for Installing Siebel Web Clients*.

When using Siebel Mobile Web Client or Developer Web Client with Google Chrome as the default browser, make sure that no other instances of Chrome are running prior to starting the Siebel client.

Closing Browser Sessions

This topic is part of *Requirements for Installing Siebel Web Clients*.

Before you install the Siebel Web Client software, it is strongly recommended that you close any browser sessions on the client computer.

For information about requirements for the Web browser environment for running Siebel Web Clients, see also *Deploying Siebel Open UI* .

Migration Installations

This topic is part of *Requirements for Installing Siebel Web Clients*.

Additional requirements apply for migration installations of the current release. For more information, see *Additional Tasks for Migration Installations*.

Requirements for Installing Siebel Tools

Review the requirements in this topic before installing Siebel Tools, as described in *About Installing Siebel Web Client or Siebel Tools*.

- You must have installed and configured the Siebel Enterprise Components software, as described in *Configuring Siebel CRM Server Modules* and *Configuring Siebel CRM Server Modules*.
- You must have created the Siebel database and installed the Siebel schema and seed data, as described in *Configuring the RDBMS* and *Installing the Siebel Database on the RDBMS*. This database stores the Siebel Tools project repositories.
- For Microsoft SQL Server deployments and Azure SQL Database deployments, valid versions of Microsoft ODBC Drivers must reside on the computer where you install the Siebel Tools Client. The Siebel Tools Client uses these drivers, but creates its own ODBC data source during the installation. Record this data source on the worksheet in *Siebel Deployment Planning Worksheet*.
- The sort order for the Siebel database for development environments must be set to binary. If it is set otherwise, then you cannot compile or merge repositories.
- Antivirus software, firewalls, and other security software can interfere with installation and operation of software if they are not properly configured. In some cases, you might need to disable antivirus software while installing Siebel CRM software. It is also recommended to exclude certain types of files from virus scanning.
- Before performing a migration installation of Siebel Tools in a deployment with multiple languages, you must make sure that the current language mode for Siebel Tools corresponds to the primary language of the Siebel database.

  > **CAUTION:** If the current language mode for Siebel Tools, before the migration installation, was not the same as the primary language, then failures will occur during the execution of Incremental Repository Merge after the migration installation. For more information about setting the language mode, see *Using Siebel Tools* . For more information about running Incremental Repository Merge, see *Siebel Database Upgrade Guide* .

- Additional requirements apply for migration installations of the current release. For more information, see *Additional Tasks for Migration Installations*.

Platform, database, and other certifications for Siebel are found on My Oracle Support. For help searching for certifications, see *How to Check the Certification Matrix in the New My Oracle Support Site* (KB871388).

Related Topics

Additional Tasks for Migration Installations

About Installing Siebel Web Client or Siebel Tools

Uninstalling Siebel CRM

11 Configuring the RDBMS

Configuring the RDBMS

This chapter provides guidelines for configuring the third-party RDBMS and creating the database instance that you will use for the Siebel database. It includes the following topics:

- *Overview of Database Configuration*
- *Configuring an Oracle Database for Siebel CRM*
- *Configuring an IBM DB2 Database for Siebel CRM*
- *Configuring a Microsoft SQL Server Database or Azure SQL Database for Siebel CRM*
- *Configuring Siebel Server Connectivity to the Siebel Database*

Overview of Database Configuration

This chapter is intended for use by database administrators (DBAs) and by others who can perform the tasks for configuring the third-party RDBMS and creating the database instance that you will use for the Siebel database.

Follow the general steps for your supported RDBMS platform, as described in this chapter. For more information, see your third-party supplier documentation, the guidelines presented in this chapter for each RDBMS, and other applicable parts of this guide.

Oracle database connector is included. Supported database versions for Oracle database are listed in certify. For other databases the database connector supported is listed and the respective supplier dictates database versions supported with that connector.

Platform, database, and other certifications for Siebel are found on My Oracle Support. For help searching for certifications, see *How to Check the Certification Matrix in the New My Oracle Support Site* (KB871388).

> **CAUTION:** *Do not perform any of the tasks mentioned in this chapter in an upgrade environment* (or in another environment where you have an existing Siebel database). In particular, you do not create the database instance on the RDBMS. However, after the upgrade is complete, you might have to modify database parameters for your RDBMS platform to match settings described in this chapter. For more information, see *About Installing in Upgrade Environments* and *Roadmap: Installing Siebel CRM in an Upgrade Case (Existing Database Requiring Full Upgrade)*.

For nonupgrade deployments (where there is no existing Siebel database), after creating the database instance and installing Siebel CRM software, you use scripts and utilities that are provided with the Database Configuration Utilities installation to create the Siebel database. Creating the Siebel database loads the Siebel schema and seed data into the database instance. For more information, see *Installing the Siebel Database on the RDBMS*.

See also *Planning RDBMS Installation and Configuration*.

In this guide, `DBSRVR_ROOT` refers to the installation directory of the Siebel Database Configuration Utilities. For more information, see *File and Directory Naming Conventions*.

Note: In general, it is easier to configure and administer a Siebel database that does not share a database instance with other applications. This approach is considered a best practice. However, some customers might decide to include multiple applications in a single database instance. The implications of this choice might differ by RDBMS type: for example, the term *database instance* has a different meaning for an Oracle Database than it does for IBM DB2. The shared database instance must be configured according to the requirements described in this chapter.

After you have completed configuring your database as described in this chapter, you can perform all of the Siebel installation and configuration tasks, including those described in:

- *Siebel CRM Download and Installation*
- *Configuring Siebel CRM Server Modules*
- *Installing the Siebel Database on the RDBMS*
- *Additional Postinstallation and Configuration Tasks*

This topic includes the following information:

- *Overview of Configuring the Database*
- *About Using Sample Scripts for Creating Siebel Database Objects*

Overview of Configuring the Database

This topic is part of *Overview of Database Configuration*.

The overall process of configuring the RDBMS and installing the Siebel database is outlined as follows. The exact process depends on your deployment requirements.

1. Install the RDBMS software:
   - Install the RDBMS server software on the appropriate server computers.
   - If using Oracle database the RDBMS client software is installed as part of Siebel. For non-Oracle databases, install the RDBMS client software on computers where you will install Siebel Server, Siebel Tools, or other modules that directly connect to the Siebel database.
2. Create the database instance.
3. Configure the parameters for the database instance.
4. Create the database, and configure the parameters for the database.
5. Configure storage settings, log space, and other elements for the Siebel database.

   Note: After you perform these tasks, you install the Siebel Enterprise Server software, including Siebel Gateway, Siebel Database Configuration Utilities, and Siebel Server. You must install Database Configuration Utilities and Siebel Server on the computer where you run the Siebel Gateway; it is optional to configure and deploy this installed Siebel Server. Then you configure the Siebel Gateway, the Siebel Enterprise, and the Siebel Application Interface profile.

6. On the computer where you installed the Database Configuration Utilities, run the Database Configuration Wizard. Perform the task to install the Siebel database (schema) in the database instance.

   This step creates the Siebel objects (tables and indexes) in the Siebel database, imports seed data for the primary language, and adds the Siebel Repository for the primary language. Also perform the other applicable tasks for the Siebel database. For more information, see *Installing the Siebel Database on the RDBMS*.

   Note: After you perform these tasks, you configure the Siebel Server, install and configure the Siebel Application Interface, and perform additional configuration tasks that are required for your deployment.

About Using Sample Scripts for Creating Siebel Database Objects

This topic is part of *Overview of Database Configuration*.

Sample scripts that are provided with the Database Configuration Utilities installation can optionally be used to create the Siebel database. *These scripts are for testing purposes in small, nonproduction environments only.* If you intend to use such a script, then first perform the tasks listed previously, through Step 6, but omitting Step 4 and Step 5.

Related Topics

Guidelines for Creating Oracle Database Objects

Guidelines for Creating IBM DB2 Database Objects

Guidelines for Creating Microsoft SQL and Azure SQL Database Server Objects

Configuring an Oracle Database for Siebel CRM

This topic includes guidelines for obtaining optimum performance from an Oracle Database. These guidelines will be useful to a broad segment of customers. Choose values for the parameters that are described in this guide that reflect the conditions in your particular environment. For more information, see Oracle Database technical documentation.

> **Note:** When scripts from the Siebel Database Configuration Utilities installation are executed in an Oracle Database, as described in *Installing the Siebel Database on the RDBMS*, sequences are generated with the CACHE option. Do not drop these sequences and regenerate them with the NOCACHE option, or else performance might be adversely affected and database contention might occur. If you require additional Oracle Database sequence objects, then you must create them manually, and you must grant SELECT on these objects to SSE_ROLE.

Collect statistics for the tables, indexes, and histograms for the columns where the data shape is skewed. Recollect these statistics whenever a large amount of data has been updated, deleted, or inserted. For more information about how to collect statistics, see the Oracle Database administration manuals.

For performance reasons, do not collect statistics for an empty table. For more information, see the Siebel Performance Guide.

Guidelines for Configuring an Oracle Database

Various kinds of guidelines are presented for configuring an Oracle Database:

- *Guidelines for Selecting a Language for Oracle Database*
- *Guidelines for Configuring Settings in the init.ora File*
- *Guidelines for Sizing Redo Logs for an Oracle Database*
- *Guidelines for Creating Oracle Database Tablespaces*
- *Guidelines for Sizing the Oracle Database*
- *Guidelines for Creating Temporary Oracle Database Tablespaces*
- *Guidelines for Overriding Oracle Database Default Tablespaces for Database Objects*

- *Guidelines for Creating Oracle Database Objects*
- *Guidelines for Ongoing Oracle Database Administration*
- *Guidelines for Using Real Application Clusters for an Oracle Database*
- *Guidelines for Optimizing Oracle Parallel Queries for Siebel Web Tools*
- *Guidelines for Installing the Siebel Database in a Pluggable Database*

Related Topics

Installing the Siebel Database on the RDBMS

Planning RDBMS Installation and Configuration

Overview of Database Configuration

Specifying the Locale for Siebel CRM

Verifying Installation for the Siebel Database

Related Books

Oracle Database SQL Language Reference

Oracle Database Administrator's Guide

Guidelines for Selecting a Language for Oracle Database

When creating your database, you must specify the character set at the database level. You specify other language characteristics at the database client level.

This topic is part of *Configuring an Oracle Database for Siebel CRM*.

To specify the character set of your database

- Execute the following command to specify the character set for your database:

```
CREATE DATABASE INSTANCE_NAME CHARACTER SET CHARACTER_SET_NAME
```

  where:

  - INSTANCE_NAME is the name of your Oracle Database instance; for example, SIEBCRM.
  - CHARACTER_SET_NAME is the textual name of the character set that you want to run; for example, WE8MSWIN1252 or AL32UTF8.

National Character Set

Siebel CRM does not use the three data types that store Unicode data using the national character set (NCHAR, NVARCHAR2, NCLOB). The default national character set, AL16UTF16, is acceptable.

Sort Order and Date Format

Follow the documented Oracle Database guidelines for client-level settings for the NLS_SORT and NLS_DATE_FORMAT parameters.

Guidelines for Configuring Settings in the init.ora File

The init.ora file contains parameters that have a major effect on the performance of Siebel CRM using Oracle Database.

Use the following settings as guidelines for your initial configuration. Your final settings will vary depending on the hardware configuration, the number of users, and the type of workload.

In the init.ora file, default parameter values are provided for small, medium, and large database deployments. Unless the configuration parameters are specified in the following settings, set them to the large database values. For detailed descriptions of each of the parameters and their effects on database performance and system resource utilization, see Oracle Database documentation.

This topic is part of *Configuring an Oracle Database for Siebel CRM*.

Brief descriptions follow for several parameters for which you might have to adjust values:

- **CURSOR_SHARING.** This parameter is set to `EXACT` by default and must not be changed.

  > **CAUTION:** Changing this value might lead to failure of some Siebel Server components.

- **DB_FILE_MULTIBLOCK_READ_COUNT.** The database buffer cache parameter dictates the number of data blocks that are read in a single Oracle Database I/O operation during a table scan. For most implementations, set this parameter to an initial value of 32. If you are using NAS storage (such as a NetApp Filer), then set the value to 8 to reduce potential network traffic problems.
- **FILESYSTEMIO_OPTION.** It is recommended to set this parameter to `SETALL`, for all operating systems. Performance issues have been observed with other settings.
- **MEMORY_TARGET.** This parameter specifies the Oracle Database system-wide usable memory. A common recommendation is to set it to a value equalling the physical memory minus 1 GB. For example, if the Oracle Database server has 16 GB of memory, then set MEMORY_TARGET to 15 GB.
- **NLS_DATE_FORMAT.** Set this parameter as needed. The default setting is `DD-MON-YY`. For information about the supported formats, see Oracle Database documentation and see *Specifying the Locale for Siebel CRM*.
- **NLS_SORT.** The sort order is specified during the initial installation of a database and defines the way in which the database sorts character data. Sort order support depends on both the code page of the database and whether it will be used in a development or a production environment. For more information, see *Specifying the Locale for Siebel CRM*.

  Considerations for development and production environments are as follows:

  - **Development environment databases.** The repository object names in your development environment database must sort using a binary sort order, because Siebel Tools uses this sort order internally. Specify

the same sort order at the database client level, so that output there does not have to be resorted. Customers are responsible for making sure their data is backed up and restored correctly.

> **Note:** Binary sort order is the simplest and fastest sort order to perform in the database. Binary sorting is case-sensitive and is based on the numeric values (for example, 0 through 255 for an 8-bit character set) of the characters in the installed character set.

- **Production environment databases.** For information about production environment database restrictions, Oracle Database documentation.

Platform, database, and other certifications for Siebel are found on My Oracle Support. For help searching for certifications, see *How to Check the Certification Matrix in the New My Oracle Support Site* (KB871388).

- **OPEN_CURSORS.** This parameter controls the amount of spaces that are reserved for the maximum number of cursors (a cursor being the same as an open query). The minimum open cursor requirement for Oracle Database is 1000 and the maximum is 2000. Within these limits, this parameter can be adjusted according to observed usage patterns.
- **OPTIMIZER_FEATURES_ENABLE.** This is an umbrella parameter for enabling a series of optimizer features that are based on an Oracle Database release number. For example, set this parameter to a value corresponding to the current release level of your Oracle Database. For more information, see Oracle Database documentation. For more information about performance tuning for Siebel CRM with an Oracle Database, see 781927.1 (Article ID) on My Oracle Support.
- **OPTIMIZER_INDEX_COST_ADJ.** It is recommended that you set this parameter to 1. Use this parameter to tune the optimizer to use index access path over a full table scan. However, depending on the data shape for your Siebel database, how statistics are gathered, and the nature of your Siebel configuration changes, other values might provide better results.
- **OPTIMIZER_MODE.** Set this parameter to `ALL_ROWS` (the default) for the Cost-Based Optimizer (CBO).

> **Note:** For more information about performance tuning for Siebel CRM with an Oracle Database, see 781927.1 (Article ID) on My Oracle Support. See also *Siebel Database Upgrade Guide* .

Guidelines for Sizing Redo Logs for an Oracle Database

If redo logs are too small, then frequent log switches occur, creating resource-intensive Oracle Database check-pointing in which all of the dirty buffers are flushed. A range of 10 to 15 minutes or longer for log switching is preferable under a normal OLTP (Online Transaction Processing) load. However, during periods of heavy DML (data manipulation language) activity, such as during large Siebel EIM loads or upgrades, the logs might switch more frequently than every two minutes. When this occurs, overall database performance suffers as a result.

You can check the frequency of this operation either in the alert log or by querying v$loghist. It is best to use verification when there is the greatest activity and the heaviest load on the database.

If this activity occurs too frequently, then drop and recreate individual redo log groups with larger sizes. A suggested minimum size is 300 MB.

This topic is part of *Configuring an Oracle Database for Siebel CRM*.

Guidelines for Creating Oracle Database Tablespaces

The initial (minimum) tablespace allocation recommendations are as follows:

- Data: 5 GB
- Index: 5 GB
- Temp: 2 GB
- System: 2 GB
- Sysaux: 1 GB

This allocation is enough for a fresh installation of Oracle Database (Unicode-enabled).

The following additional guidelines will help you in creating tablespaces:

- To improve performance on your production system, create at least two tablespaces for a Siebel implementation: one for indexes and one for data.
- Distribute objects that you anticipate to be large or points of contention by creating additional separate tablespaces (preferably on separate disk devices).
- Be sure that you, or whoever is responsible for setting up permissions, grant the Siebel table owner account the privilege and sufficient quota to create tables and indexes in these tablespaces.

  Besides the table owner, the database user ID used for Siebel Marketing also requires additional rights at the database level within the OLTP schema. You must grant DROP TABLE, DROP INDEX, CREATE TABLE, and CREATE INDEX rights to this user. For more information, see *Siebel Marketing Installation and Administration Guide* .
- Set storage parameters for your data and index tablespaces. The Siebel database installation procedure does not set storage parameters for the objects it creates. The storage configuration for each object follows the default storage parameters of its tablespace. It is recommended that you create locally managed tablespaces by using the following syntax:

  ```
  extent management local autoallocate segment space management auto;
  ```

- In a development or test environment, multiple Siebel CRM installations can coexist on one Oracle Database instance. Install each Siebel database under a separate table owner, so that each schema owner is unique.
- Function-based indexes based on expressions that require QUERY_REWRITE_ENABLED to equal TRUE are not supported. However, DESC (descending) indexes are supported, as in a standard schema.

This topic is part of *Configuring an Oracle Database for Siebel CRM*.

Guidelines for Sizing the Oracle Database

Monitor object growth and fragmentation carefully and alter the database storage parameters as required. Also use the following guidelines for initial sizing of your Oracle Database. (If you use EXTENT MANAGEMENT LOCAL AUTOALLOCATE for CREATE DATABASE, then these guidelines *do not apply*.)

- Set the initial extent to a very small size (the minimum is one database block), so that empty tables and indexes do not consume large amounts of space. For example, start with either two or four blocks (in other words, 16 KB or 32 KB with an 8-KB block size). This allocation promotes less fragmentation.

Even if you have as many 10,000 objects, this number of objects uses only 312 MB, which is far less space required than for some standard office software packages.

- Set the default next extent for your data and index tablespaces to a minimum of 100 KB.

This topic is part of *Configuring an Oracle Database for Siebel CRM*.

Guidelines for Creating Temporary Oracle Database Tablespaces

Modify all of the user temporary tablespace definitions from the default of `SYSTEM` to the name of the temporary tablespace; for example, `TEMP`.

To find out which users are assigned to which temporary tablespaces, query the TEMPORARY_TABLESPACE column of DBA_USERS. If any users are assigned to a tablespace other than the one allocated for temporary sort-type operations, then correct the situation.

This topic is part of *Configuring an Oracle Database for Siebel CRM*.

Guidelines for Overriding Oracle Database Default Tablespaces for Database Objects

Siebel CRM provides the option of overriding the default storage parameters for the tablespaces in which specific tables or indexes are created. You created these tablespaces by using the instructions in *Guidelines for Creating Oracle Database Tablespaces*. To override these parameters, edit the ddl.ctl file located in the `DBSRVR_ROOT\oracle` directory.

Note: The ddl.ctl file must not be modified except by a qualified DBA.

For each Siebel object (table or index), you can specify a tablespace by using the Table Space parameter. In the following example, the tablespace for the table S_APP_VIEW is set to `DATA1`.

This topic is part of *Configuring an Oracle Database for Siebel CRM*.

As provided by Siebel CRM, the .ctl file does not set storage parameters for the objects it creates, so that they default to the parameters of the tablespaces in which they are created. However, the Table Space parameter works only under the following conditions:

- When the table does not yet exist (for example, when you are performing a new database installation).
- When the table must be rebuilt. In other words, when schema changes are made to the table such that an ALTER TABLE command is insufficient to implement the schema changes. In this case, the table must be dropped and recreated.

The following example illustrates the use of the Table Space parameter to set storage values for specific tables:

```
[Object 219]
Type = Table
Name = S_APP_VIEW
Column 1 = ROW_ID VARCHAR(15) NOTNULL
Column 2 = CREATED TIMESTAMP NOTNULL DEFAULT %NOW%
Column 3 = CREATED_BY VARCHAR(15) NOTNULL
Column 4 = LAST UPD TIMESTAMP NOTNULL DEFAULT %NOW%
Column 5 = LAST_UPD_BY VARCHAR(15) NOTNULL
```

```
Column 6 = DCKING_NUM NUMERIC(22,7) DEFAULT 0
Column 7 = MODIFICATION_NUM NUMERIC(10,0) NOTNULL DEFAULT 0
Column 8 = CONFLICT_ID VARCHAR(15) NOTNULL DEFAULT '0'
Column 9 = NAME VARCHAR(50) NOTNULL
Column 10 = DESC_TEXT VARCHAR(255)
Column 11 = LOCAL_ACCESS_FLG CHAR(1)
Table Space = data1
```

If you use locally managed tablespaces and want to change the storage parameters, then see Oracle Database technical documentation.

For an example (IBM DB2) of overriding the defaults for specific tables and indexes, see *Guidelines for Overriding IBM DB2 Default Tablespaces for Database Objects*.

Guidelines for Creating Oracle Database Objects

Siebel CRM provides sample scripts (CrBlankOracleSiebelDEMO.sql and CrBlankOracleSiebelDEMOPostCrDB.sql), located in the `DBSRVR_ROOT\Oracle` directory. Use these scripts as a reference to help you create your own scripts for creating a blank Siebel database, based on the requirements for your deployment.

Use the settings in the script as guidelines for your initial configuration. Your final settings will vary depending on the server hardware configuration, the number of users, and the type of workload. Use a small, nonproduction environment for testing purposes.

Before using these scripts, read the file CrBlankOracleSiebelDEMOReadMe.txt. See also *Overview of Database Configuration*.

After you install the Database Configuration Utilities on the Siebel Server computer, as described in *Siebel CRM Download and Installation*, you can modify the database table and index creation scripts to specify the tablespace names that you created for Siebel tables and indexes. For more information, see *Guidelines for Overriding Oracle Database Default Tablespaces for Database Objects*.

Additional information about Oracle Database configuration and tuning options is available from Oracle, your hardware vendor, and other sources.

This topic is part of *Configuring an Oracle Database for Siebel CRM*.

Guidelines for Ongoing Oracle Database Administration

After your Siebel CRM installation is up and running, monitor the following areas on a regular basis:

- **Insertion rates on tables.** You probably will want to set the INI_TRANS value for tables with high insertion rates to a value higher than 1; a typical setting is 4.
  This parameter determines how many simultaneous inserts can occur on the database blocks that store data for those tables and, therefore, can affect performance in an intensive data-entry environment. Use multiple freelists for the table S_DOCK_TXN_LOG, because this table receives numerous inserts.
- **SGA cache hits.** Determine whether SGA parameters have to be adjusted for your system.
- **The extents used by each object.** A large number of extents on a table or index creates response time degradation for transactions that access the table or index.
- **Siebel tables that are subject to frequent INSERT and DELETE operations.** This transaction mixture can cause some database tables to become fragmented over time.

If you are using Siebel Remote, then your DBA must monitor space utilization and fragmentation of the following tables, and perform regular database maintenance procedures as recommended for an Oracle Database. Monitor the following tables in particular, because they will have frequent changes when transaction logging is enabled:

- S_SRM_REQUEST
- S_DOCK_TXN_LOG
- S_DOCK_TXN_LOGT
- S_DOCK_TXN_SET
- S_DOCK_TXN_SETT
- S_DOCK_INST
- S_DOCK_INIT_ITEM

Your DBA might also choose to monitor all of the tables and indexes in the Siebel schema, reorganizing them when required.

This topic is part of *Configuring an Oracle Database for Siebel CRM*.

Guidelines for Using Real Application Clusters for an Oracle Database

Siebel CRM supports Oracle Real Application Clusters (RAC) failover configurations for Oracle Database. Both active-passive and active-active RAC configurations are supported.

For more information, see:

- 473859.1 (Article ID) on My Oracle Support. This document was previously published as Siebel Technical Note 635.
- 478215.1 (Article ID) on My Oracle Support. This document was previously published as Siebel FAQ 2220.
- Concepts documentation for Oracle Real Application Clusters software on Oracle Technology Network.

This topic is part of *Configuring an Oracle Database for Siebel CRM*.

Guidelines for Optimizing Oracle Parallel Queries for Siebel Web Tools

Several queries performed by Siebel Web Tools or Siebel Tools can benefit from the parallel query feature for Oracle Database. Parallel hints have been added to the `SELECT` statements for such queries for Siebel Web Tools and Siebel Tools. The Siebel component parameter Oracle Degree of Parallelism (alias OraDegreeOfParallelism) allows setting the degree of parallelism for Siebel Web Tools or Siebel Tools, where parallel hints are present in the `SELECT` statements.

By default, the setting of OraDegreeOfParallelism is `auto`, which requires parallel settings in the Oracle Database, and might affect development work in other areas.

To optimize parallel queries, it is recommended to set OraDegreeOfParallelism for the Siebel Web Tools component to a value of `4`. To do so, enter a command like the following in Server Manager:

```
change param OraDegreeOfParallelism=4 for comp SWToolsObjMgr_enu
```

To set this parameter for Siebel Tools, add OraDegreeOfParallelism under the [Siebel] section in tools.cfg, and set it to `4`.

> **Note:** The parameter Oracle Optimizer Ignore Hints (alias OraOptimizerIgnoreHints) is related. For parallel queries, do not change the default value, `False`, to `True`. If OraOptimizerIgnoreHints is set to `True`, then OraDegreeOfParallelism has no effect.

For more information using Server Manager and about setting component parameters, see *Siebel System Administration Guide* .

This topic is part of *Configuring an Oracle Database for Siebel CRM*.

Guidelines for Installing the Siebel Database in a Pluggable Database

You can install the Siebel database in a multitenant environment with versions of Oracle Database that support this feature. To do so, first create a pluggable database within the container database. Then create a corresponding entry for the pluggable database in tnsnames.ora. This entry must specify the service you will use to connect to the pluggable database.

When you run grantusr.sql, you do so in the pluggable database rather than in the container database. Database settings you specify in the Siebel Management Console must be those for the pluggable database and you must create the Siebel schema within the pluggable database.

For more information about creating a pluggable database and about managing a multitenant environment, see Oracle Database documentation on Oracle Help Center.

This topic is part of *Configuring an Oracle Database for Siebel CRM*.

Configuring an IBM DB2 Database for Siebel CRM

This topic includes guidelines for obtaining optimal performance from an IBM DB2 Database for use with Siebel CRM. These guidelines will be useful to a broad segment of customers. Choose values for the parameters described in this guide that reflect conditions in your particular environment. For more information, see IBM DB2 technical documentation.

> **Note:** In this guide, the terms *IBM DB2* or *DB2* are often used to refer to the database platform IBM DB2 for Linux, UNIX, and Windows.

When you use IBM DB2, the IBM Data Server Client must be installed on the Siebel Server computer. Verify that the IBM Data Server Client is installed before proceeding. For details, see *Siebel Database Upgrade Guide* . In addition, the IBM Data Server Runtime Client must be installed on the Siebel Server computer.

Platform, database, and other certifications for Siebel are found on My Oracle Support. For help searching for certifications, see *How to Check the Certification Matrix in the New My Oracle Support Site* (KB871388).

> **Note:** If your database is IBM DB2 for z/OS, then see *Implementing Siebel Business Applications on DB2 for z/OS* instead of this chapter.

Guidelines for Configuring an IBM DB2 Database

Various kinds of guidelines are presented for configuring an IBM DB2 database:

- *Guidelines for Setting IBM DB2 Database Configuration Parameters*
- *Guidelines for Selecting a Language for IBM DB2*
- *Guidelines for Creating the IBM DB2 Database*
- *Guidelines for Setting IBM DB2 Configuration Parameters*
- *Guidelines for Setting Up IBM DB2 Buffer Pools*
- *Guidelines for Creating IBM DB2 Tablespaces*
- *Guidelines for Overriding IBM DB2 Default Tablespaces for Database Objects*
- *Guidelines for Preventing IBM DB2 from Running Out of ODBC Statement Handles*
- (UNIX) *Guidelines for Increasing the Number of IBM DB2 User Processes Under AIX*
- *Guidelines for Determining IBM DB2 Log Space*
- *Guidelines for Creating IBM DB2 Database Objects*
- *Guidelines for Managing IBM DB2 Fragmentation*

Related Topics

Installing the Siebel Database on the RDBMS

Planning RDBMS Installation and Configuration

Overview of Database Configuration

Specifying the Locale for Siebel CRM

Verifying Installation for the Siebel Database

Related Books

Siebel Database Upgrade Guide

Implementing Siebel Business Applications on DB2 for z/OS

Guidelines for Setting IBM DB2 Database Configuration Parameters

This topic is part of *Configuring an IBM DB2 Database for Siebel CRM*.

You can set the IBM DB2 database configuration parameters by using the `update database manager configuration` command of the DB2 command line processor or DB2 command prompt (Windows). For more information about modifying these configuration parameters, see IBM DB2 technical documentation.

IBM DB2 Database Configuration Parameters

The following table describes IBM DB2 database configuration parameters that differ from the default settings. Set these parameters for each DB2 instance. Use the configuration information in the following table for the listed parameters. For parameters not listed in this table, accept the default settings.

Parameter	Explanation	Setting or Comment
SHEAPTHRES	Sort heap threshold (4 KB)	`200000` Deployments with 3,000 or more concurrent users and using over 5 GB of RAM can increase this to `300000`. If you are using automatic management, then set this parameter to 0.
DIR_CACHE	Directory cache support	`YES`
ASLHEAPSZ	Application support layer heap size	`15`
RQRIOBLK	Maximum requester I/O block size (bytes)	`65535`
MON_HEAP_SZ	Database monitor heap size (4 KB)	`128` (minimum)
KEEPFENCED	Keep Fenced process	`YES`
NUM_INITAGENTS	Initial number agents in pool	`10`
NUM_POOLAGENTS	Number of agents in the agent pool kept active at all times	`80`
MAX_COORDAGENTS	Maximum number coordinating agents	`automatic`
INDEXREC	Index recreation time	`RESTART`
INTRA_PARALLEL	Enable intra-partition parallelism	`NO`
INSTANCE_MEMORY	Amount of memory to be reserved for instance management	`automatic`

DB2set Parameters

Use the `db2set` command to set the parameters (for example, `db2set DB2_HASH_JOIN = NO`) referenced in the following table. On Windows, you access this command through the DB2 command prompt (db2cmd).

Parameter	Explanation	Setting
DB2_PARALLEL_IO	Useful when using RAID devices. For more information, see IBM DB2 documentation.	* (asterisk)
DB2_REDUCED_OPTIMIZATION	Controls optimization techniques used at specific optimization levels.	YES

Guidelines for Selecting a Language for IBM DB2

This topic is part of *Configuring an IBM DB2 Database for Siebel CRM*.

As part of database creation, you must set the language characteristics of your database, even if you deploy in only one language.

To do this, you must know in which of the Siebel-supported languages your database runs, the codeset your database uses, the territory for your language, and the sort order (also known as the collation sequence) that your users prefer.

For an IBM DB2 production environment database, you can use any sort order. For a development environment database, you must use binary (identity) sort order.

Setting the language characteristics of the database is part of the sample script in the `DBSRVR_ROOT\DB2UDB` directory.

For Siebel CRM language support and Unicode support, see the Siebel Global Deployment Guide.

See also *Planning RDBMS Installation and Configuration* and *Specifying the Locale for Siebel CRM*.

See also *Verifying Installation for the Siebel Database*.

Codeset

IBM DB2 distinguishes between a code page (also known as a character set) and a codeset. A *codeset* is defined as a textual string that describes the character encoding standard used for the database, whereas a *code page* is a numeric representation of the same standard.

Territory

The *territory*, or region, is a combination of the language and the locale; for example, French would be a language example, while Canada or France would be locales in which French is used with regional differences. So, an example of a territory is Canadian French.

Sort Order

The sort order is specified during the initial installation of a database and defines the way in which the database sorts character data. Sort order support depends on both the code page of the database and whether it will be used in a development or a production environment.

- **Development environment databases.** The repository object names in your development environment database must sort in the same order that they would under the UTF-16 binary sort order, because Siebel Tools uses this sort order internally.

  > **Note:** Binary sort order is the simplest and fastest sort order to perform in the database. Binary sorting is case-sensitive and is based on the numeric values (for example, 0 through 255 for an 8-bit character set) of the characters in the installed character set.

  Customers are responsible for making sure that their data is backed up and restored correctly.
- **Production environment databases.** For information about production environment database restrictions, see the IBM DB2 documentation.

Platform, database, and other certifications for Siebel are found on My Oracle Support. For help searching for certifications, see *How to Check the Certification Matrix in the New My Oracle Support Site* (KB871388).

Guidelines for Creating the IBM DB2 Database

This topic is part of *Configuring an IBM DB2 Database for Siebel CRM*.

If you are installing a Unicode database for IBM DB2, then you must specify `UTF-8` as the codeset (including the hyphen). UTF-8 is the parameter used for Unicode implementations on IBM DB2, although the processing will use UCS-2. When you specify UTF-8 as the encoding for the VARCHAR type, the encoding for the VARGRAPHIC type is automatically set to UCS-2, even though UCS-2 is not specified as the parameter.

Verify that your data is exported and imported correctly.

For Siebel CRM language support and Unicode support, see the Siebel Global Deployment guide.

Platform, database, and other certifications for Siebel are found on My Oracle Support. For help searching for certifications, see *How to Check the Certification Matrix in the New My Oracle Support Site* (KB871388).

To create the IBM DB2 database

1. Locate the primary (base) language that your database will use, the territory for your language, and the applicable codeset.
2. Using the DB2 command line processor (Unix) or DB2 command prompt (Windows), enter the following command:

   ```
   db2 create database dbname using codeset
    territory collate using identity
   ```

   where:
   - dbname is the alias for your database
   - codeset is the textual representation of your code page

- territory is the territory for the language that your database runs in, under that codeset

Guidelines for Setting IBM DB2 Configuration Parameters

This topic is part of *Configuring an IBM DB2 Database for Siebel CRM*.

The database configuration parameters can be set by using the `update database configuration` command of the DB2 command line processor or DB2 command prompt (Windows).

For more information about modifying these configuration parameters, see IBM DB2 technical documentation.

The following table describes IBM DB2 database configuration parameters that differ from the default settings. However, these descriptions are guidelines only. Set these parameters for each database within an instance on which you run your Siebel application. For other parameters, accept the default settings.

Parameter	Explanation	Setting
DFT_DEGREE	Degree of parallelism (the value 1 turns query parallelism off).	`1`
DFT_QUERYOPT	Default query optimization class. This parameter only takes effect on the database server and affects the Siebel Server components, such as Siebel EIM or Siebel Remote. Queries run through the UI are not affected by this setting. They take the value of the Siebel system preference DB2: Default Opt Level, or you can override them at the business component level by using Siebel Tools.	`3`
DBHEAP	Database heap (4 KB).	Automatic.
CATALOGCACHE_SZ	Catalog cache size (4 KB).	`10000`
LOGBUFSZ	Log buffer size (4 KB).	`512`
LOCKLIST	Maximum storage for lock list (4 KB).	Automatic.
SORTHEAP	Sort list heap (4 KB). Use lower values for development environments; use higher values for production. However, increasing this value can lead to insufficient memory on the database server. Also, this parameter might have to be set below the recommended range if you have a high number of Siebel users. Always monitor database server memory and performance to find the best setting for your environment.	Automatic.
STMTHEAP	Minimum setting. If necessary, increment this parameter in 1048 blocks of 4 KB.	Automatic or a fixed value.

Parameter	Explanation	Setting
		For example, you might update the parameter setting using a command like this: `DB2 UPDATE DB CFG FOR SAMPLE USING STMTHEAP 8192 AUTOMATIC` This value specifies 8192 times 4 KB (or 32 MB) for dynamic join enumeration and unlimited for greedy join enumeration.
STAT_HEAP_SZ	Statistics heap size (4 KB).	Automatic.
MAXLOCKS	Percentage of lock lists for each application.	Automatic.
LOCKTIMEOUT	Lock time out (seconds).	`300`
CHNGPGS_THRESH	Changed pages threshold.	`60`
NUM_IOCLEANERS	Number of asynchronous page cleaners.	Automatic.
SEQDETECT	Sequential detect flag.	`YES`
DFT_PREFETCH_SZ	Default prefetch size (4 KB).	`32`
MAXAPPLS	Maximum number of active applications.	Automatic.
AVG_APPLS	Average number of active applications.	Automatic.
MAXFILOP	Maximum DB files open for each application.	`500`
LOGFILSIZ	Log file size (in 4 KB increments).	`65535`
LOGPRIMARY	Number of primary log files.	`25` to `50` The value of LOGPRIMARY and LOGSECOND together must not exceed 256.
LOGSECOND	Number of secondary log files.	Up to `103` The value of LOGPRIMARY and LOGSECOND together must not exceed 256.

Parameter	Explanation	Setting
SOFTMAX	Percent log file reclaimed before soft checkpoint.	80
APPLHEAPSZ	Default application heap (4 KB).	Automatic.
PCKCACHESZ	Package cache size (4 KB).	Automatic.
NUM_IOSERVERS	Number of disks on which the database resides.	Automatic.

Guidelines for Setting Up IBM DB2 Buffer Pools

This topic is part of *Configuring an IBM DB2 Database for Siebel CRM*.

A *buffer pool* is an area of main system memory that is used for holding pages of data that have been fetched from the tablespace. In IBM DB2, each tablespace is associated with a buffer pool. Adding more space to a buffer pool enhances the performance of the database.

You must have at least three buffer pools for the Siebel tablespaces. You can use the default buffer pool to buffer data pages from all of the Siebel 4-KB tablespaces.

You must also create additional buffer pools with 16-KB and 32-KB page sizes for sorting and other SQL processing. A sample configuration is shown in the following table.

Buffer Pool	Suggested Buffer Pool Size	Page Size
BUF4K	50% of available memory	4 KB
BUF16K	25% of available memory	16 KB
BUF32K	32 MB	32 KB

Different operating systems support different maximum amounts of IBM DB2 addressable memory. Depending on the memory configuration of a given server, the suggested pool sizes for BUF4K and BUF16K buffer pools might exceed these maximums, requiring you to allocate a smaller percentage. To determine the optimal buffer pool sizes, use IBM DB2 monitoring features.

About Buffer Pools and the Full Publish Process

When the Full Publish process runs to generate Siebel Runtime Repository data from the Design Repository (including during the upgrade process for the development database), it makes intensive use of workspace queries, which results in a higher usage of buffer pools. If you encounter errors similar to the following during the Full Publish process, increase the buffer pool sizes above the values shown in the preceding table and try again.

```
DBCLog DBCLogError 1 000000025c1eca2c:0 2018-12-22 04:50:04 SQLError:
sqlstate 57011: [IBM][CLI Driver][DB2/AIX64] SQL1218N There are no pages
```

```
currently available in bufferpool "4". SQLSTATE=57011
```

For more information about the Full Publish process, see *Using Siebel Tools* and *Siebel Database Upgrade Guide* .

Guidelines for Creating IBM DB2 Tablespaces

This topic is part of *Configuring an IBM DB2 Database for Siebel CRM*.

The Siebel database installation process described in *Installing the Siebel Database on the RDBMS* specifies the tablespaces in which to store your Siebel tables and indexes.

A Siebel database on IBM DB2 requires at least four tablespaces using database-managed space (DMS). Each tablespace can have one or more tablespace containers to store the data. Create a *minimum* of four DB2 tablespaces to hold your tables and indexes: a 4-KB, a 16-KB, and a 32-KB tablespace, for your various sized tables, and an additional tablespace to hold your indexes. The tablespaces must be created as database-managed space. Use a small, nonproduction environment for testing purposes.

Observe the following guidelines when creating tablespaces:

- Create at least three IBM DB2 tablespaces for tables of various sizes, as shown in the table that follows this list. Using the default tablespace names is recommended.
- Create additional tablespaces as required for individual tables, such as S_DOCK_TXN_LOG. If you expect to have large, heavily used tables, then put these in their own tablespace.
- In addition, for performance reasons it is highly recommended that you create a separate tablespace for indexes. For example, you might create a tablespace named SIEBEL_IDX and using buffer pool BUF16K, with a page size of 16 KB, and with 3 GB allocated for this tablespace.
- Create at least 4-KB, 16-KB, and 32-KB temporary tablespaces to use for sorting and other SQL processing as described in the following topics. If you do not create them, then your database will experience serious performance and stability problems. Use system-managed space (SMS) for all of the temporary tablespaces. Make sure that these temporary tablespaces are expandable to 2 GB for storage purposes.
- If you intend to use the DB2 Load utility to populate EIM tables, then this method makes the tablespace in which EIM tables resides unavailable for the duration of the load. Placing the EIM tables in one or more separate tablespaces allows concurrent activity on the database while the load utility is running.
- To override default storage parameters, such as the tablespace definitions, see *Guidelines for Overriding IBM DB2 Default Tablespaces for Database Objects*.
- Record the tablespace names in *Siebel Deployment Planning Worksheet*.

DB2 Tablespace Name	Buffer Pool Name	Recommended Value	Description
Non-Unicode Database			
SIEBEL_4K	BUF4K	2 GB	Tablespace name for tables with row sizes less than 4006 bytes.
SIEBEL_16K	BUF16K	300 MB	Tablespace name for tables with row sizes from 4006 bytes through 16,293 bytes.

DB2 Tablespace Name	Buffer Pool Name	Recommended Value	Description
SIEBEL_32K	BUF32K	100 MB	Tablespace name for tables with row sizes greater than 16,293 bytes.
Unicode-Enabled Database			
SIEBEL_4K	BUF4K	4 GB	Tablespace name for tables with row sizes less than 4006 bytes.
SIEBEL_16K	BUF16K	700 MB	Tablespace name for tables with row sizes from 4006 bytes through 16,293 bytes.
SIEBEL_32K	BUF32K	100 MB	Tablespace name for tables with row sizes greater than 16,293 bytes.

Guidelines for Overriding IBM DB2 Default Tablespaces for Database Objects

This topic is part of *Configuring an IBM DB2 Database for Siebel CRM*.

Siebel CRM provides the option of overriding the default storage parameters for the tablespaces in which specific tables or indexes are created. You created these tablespaces by using the instructions in *Guidelines for Creating IBM DB2 Tablespaces*. To override these defaults, edit the ddl.ctl file located in the `DBSRVR_ROOT\DB2UDB` directory.

Note: The ddl.ctl file must not be modified except by a qualified DBA.

For each Siebel table, you can specify a tablespace by using the Table Space parameter. In the following example, the tablespace for the table S_APP_VIEW is set to `DATA1`.

As provided, the .ctl file does not set storage parameters for the objects it creates, so they default to the parameter settings of the tablespaces in which they are created. However, the Table Space parameter works only under the following conditions:

- When the table does not yet exist (for example, when you are performing a new database installation).
- When the table must be rebuilt. In other words, when schema changes are made to the table such that an ALTER TABLE command is insufficient to implement the schema changes. In this case, the table must be dropped and recreated.

As shown in the following example, you can use the Table Space parameter to set storage parameters for specific tables:

```
[Object 219]
Type = Table
Name = S_APP_VIEW
Column 1 = ROW_ID VARCHAR(15) NOTNULL
Column 2 = CREATED TIMESTAMP NOTNULL DEFAULT %NOW%
Column 3 = CREATED_BY VARCHAR(15) NOTNULL
Column 4 = LAST_UPD TIMESTAMP NOTNULL DEFAULT %NOW%
Column 5 = LAST_UPD_BY VARCHAR(15) NOTNULL
```

```
Column 6 = DCKING_NUM NUMERIC(22,7) DEFAULT 0
Column 7 = MODIFICATION_NUM NUMERIC(10,0) NOTNULL DEFAULT 0
Column 8 = CONFLICT_ID VARCHAR(15) NOTNULL DEFAULT '0'
Column 9 = NAME VARCHAR(50) NOTNULL
Column 10 = DESC_TEXT VARCHAR(255)
Column 11 = LOCAL_ACCESS_FLG CHAR(1)
Table Space = data1
```

The following example illustrates how to override the defaults for specific tables and indexes:

```
[Object 7135]
Type = Table
Name = S_EVT_ACT
Group = Activity-1
Append Mode = Yes
Column 1 = ROW_ID WVARCHAR(15) NOTNULL
Column 2 = CREATED TIMESTAMP NOTNULL DEFAULT %NOW%
Column 3 = CREATED_BY WVARCHAR(15) NOTNULL
Column 4 = LAST_UPD TIMESTAMP NOTNULL DEFAULT %NOW%
Column 5 = LAST_UPD_BY WVARCHAR(15) NOTNULL
Column 6 = DCKING_NUM NUMERIC(22,7) DEFAULT 0
Column 7 = MODIFICATION_NUM NUMERIC(10,0) NOTNULL DEFAULT 0
Column 8 = CONFLICT_ID WVARCHAR(15) NOTNULL DEFAULT '0'
Column 9 = ACTIVITY_UID WVARCHAR(30) NOTNULL DEFAULT 'x'
...
Column 166 = TODO_CD WVARCHAR(30)
Column 167 = USER_MSG_ID WVARCHAR(15)
Column 168 = WC_START_VIEW WVARCHAR(250)
Column 169 = WC_TYPE_CD WVARCHAR(30)

[Object 7136]
Type = Index
Name = S_EVT_ACT_F1
Table = S_EVT_ACT
Column 1 = CON_PRDINT_ID ASC
Index Space = S_EVT_ACT_TBS_IDX

[Object 7137]
Type = Index
Name = S_EVT_ACT_F10
Table = S_EVT_ACT
Allow Reverse Scans = Yes
Column 1 = TARGET_OU_ID ASC
Column 2 = APPT_START_DT DESC
Column 3 = ROW_ID ASC

[Object 7138]
Type = Index
Name = S_EVT_ACT_F11
Table = S_EVT_ACT
Column 1 = PAR_EVT_ID ASC
Index Space = S_EVT_ACT_TBS_IDX
```

Guidelines for Preventing IBM DB2 from Running Out of ODBC Statement Handles

This topic is part of *Configuring an IBM DB2 Database for Siebel CRM*.

IBM DB2 can quickly run out of ODBC statement handles, depending on the number of business objects your enterprise uses. Because it is difficult to know how many business objects your users use, increase this number automatically each

time that you install the DB2 Client, or when you rebind database utilities. You can increase the number of CLI packages with the CLIPKG option for the bind command.

To rebind the CLI packages

1. (Windows) Navigate to sqllib\bnd (C:\sqllib\bnd from a DB2 Command Window) in the IBM DB2 instance home directory, by using a method appropriate to your operating system.
2. (UNIX) Navigate to sqllib/bnd in the IBM DB2 instance home directory, by using a method appropriate to your operating system.
3. Connect to the DB2 database, and enter a command similar to the following example, which sets the number of CLI packages to 30:

   ```
   db2 bind @db2cli.lst blocking all grant public clipkg 30
   ```

   Specify a value for CLIPKG that is sufficient to run your applications. For more information about the DB2 bind command and the CLIPKG option, see IBM DB2 documentation.

Guidelines for Increasing the Number of IBM DB2 User Processes Under AIX

This topic is part of *Configuring an IBM DB2 Database for Siebel CRM*.

When you reach about 400 concurrent database users running under AIX, the connection to IBM DB2 might fail. To avoid this problem, reset the parameter controlling the maximum number of user processes on your AIX server.

To reset the parameter

1. Log on to the server as the AIX system administrator.
2. Navigate to $SIEBEL_HOME (the Siebel Server root directory) and source environment variables, by using one of the following commands, depending on the type of shell that you use:

   C shell

   ```
   source siebenv.csh
   ```

   Bourne or Korn shell

   ```
   . ./siebenv.sh
   ```

   > **Note:** Make sure that there is a space between the initial period and ./siebenv.sh.

3. Execute the following command to review the options:

   ```
   smitty chgsys
   ```

4. Select Maximum Number of Processes and reset the default (512) to a larger number; for example, specify a value like 10,000 to avoid imposing an upper limit on the number of processes that a single user can spawn.

Guidelines for Determining IBM DB2 Log Space

This topic is part of *Configuring an IBM DB2 Database for Siebel CRM*.

You must create database transaction log files large enough to support various large transactions used by Siebel CRM. On IBM DB2, three parameters affect the amount of log file space reserved:

- **LOGFILSIZ.** The size of the log file.
- **LOGPRIMARY.** The number of log files to preallocate and use.
- **LOGSECOND.** Extra log files that are allocated only if they are needed for a large transaction.

For a large system, allocate 4 to 8 GB of total log file space, as needed. Create 25 to 50 primary log files of 160 MB each, by setting the LOGFILSIZ database configuration parameter to `40000` and the LOGPRIMARY parameter to a value between `25` to `50`. To support very large transactions, set the LOGSECOND parameter to 128 minus the value of LOGPRIMARY. Smaller systems can use less log file space.

Guidelines for Creating IBM DB2 Database Objects

This topic is part of *Configuring an IBM DB2 Database for Siebel CRM*.

Siebel CRM provides a sample script (CrBlankDB2UDBSiebelDEMO.sql), located in the `DBSRVR_ROOT\DB2UDB` directory. Use this script as a reference to help you create your own scripts for creating a blank Siebel database based on the requirements for your deployment.

Use the settings in the script as guidelines for your initial configuration. Your final settings will vary depending on the server hardware configuration, the number of users, and the type of workload. Use a small, nonproduction environment for testing purposes.

Before using this script, read the file CrBlankDB2UDBSiebelDEMOReadMe.txt. See also *Overview of Database Configuration*.

After you install the Database Configuration Utilities on the Siebel Server computer, as described in *Siebel CRM Download and Installation*, you can modify the database table and index creation scripts to specify the tablespace names that you created for Siebel tables and indexes. For more information, see *Guidelines for Overriding IBM DB2 Default Tablespaces for Database Objects*.

Additional information about IBM DB2 configuration and tuning options is available from IBM, your hardware vendor, and other sources.

Guidelines for Managing IBM DB2 Fragmentation

This topic is part of *Configuring an IBM DB2 Database for Siebel CRM*.

No strict guidelines can be offered as to which tables and indexes might be fragmented, because of the variety in application and customer operation variables at any given customer site. However, DBAs must pay attention to the status of large or heavily used tables, because fragmentation of these tables can affect performance significantly. For a list of these Siebel tables, see *Siebel Deployment Planning Guide* .

Note: Do not reorganize S_ESCL_LOG, S_DOCK_INIT_ITEM, S_ESCL_ACTN_REQ, S_APSRVR_REQ, and all tables named S_DOCK_INITM_%% (in this table name, % is a digit), because these tables are defined to be in append mode.

Use the following strategy to manage table fragmentation:

- Reorganize any tables, as needed, by running `REORG TABLE`. For details on how to reorganize tables or indexes, see 477378.1 (Article ID) on My Oracle Support. This document was previously published as Siebel FAQ 2072. After reorganizing tables, update statistics by using the `runstats` utility on any reorganized tables with the following minimum parameters:

  ```
  runstats on table tablename with distribution and detailed indexes all shrlevel change
  ```

  You might add other parameters as required, but use the shrlevel change parameter to allow concurrent access to your tables while `runstats` executes.

  CAUTION: The `runstats` utility overwrites statistics loaded by Siebel CRM. If you use `runstats`, then always execute loadstats.sql afterwards, by using either the DB2 command line processor or `odbcsql`. Otherwise, valuable statistics will be lost.

  To run loadstats.sql by using `odbcsql`, use the following command:

  ```
  odbcsql /s DATASOURCE_NAME /u username /p password /v separator
   siebel_root/dbsrvr/db2udb/loadstats.sql TABLEOWNER_NAME
  ```

Configuring a Microsoft SQL Server Database for Siebel CRM

This topic includes guidelines for obtaining optimal performance from the Microsoft SQL Server database for use with Siebel CRM. These guidelines will be useful to a broad segment of customers. Choose values for the parameters described in this guide that reflect conditions in your particular environment. For more information, see Microsoft SQL Server technical documentation.

Note: Customers using Azure SQL Databases should follow the Microsoft SQL Server Database recommendations in this section unless otherwise noted.

Guidelines for Configuring a Microsoft SQL Server Database

Various kinds of guidelines are presented for configuring a Microsoft SQL Server database:

- *Guidelines for Configuring Microsoft SQL Server Parameters*
- *Guidelines for Selecting a Language for Microsoft SQL Server*
- *Guidelines for Creating the Microsoft SQL Server Database*
- *Guidelines for Allocating Microsoft SQL Server Database Log Space*
- *Guidelines for Overriding Microsoft SQL Server Default Tablespaces for Database Objects*

- *Guidelines for Creating Microsoft SQL and Azure SQL Database Server Objects*
- *Guidelines for Ongoing Microsoft SQL Server Administration*
- *Updating Microsoft SQL Server Statistics*
- *Managing Microsoft SQL Server Fragmentation*

Related Topics

Installing the Siebel Database on the RDBMS

Planning RDBMS Installation and Configuration

Overview of Database Configuration

Specifying the Locale for Siebel CRM

Verifying Installation for the Siebel Database

Guidelines for Configuring Microsoft SQL Server Parameters

This topic is part of *Configuring a Microsoft SQL Server Database or Azure SQL Database for Siebel CRM*.

Guidelines for setting the Microsoft SQL Server parameters for maximum performance follow. For more information, see Microsoft SQL Server technical documentation.

Review the descriptions of the following parameters and reset the values as appropriate to your deployment:

- **max degree of parallelism.** This parameter specifies whether query plans are generated for parallel execution on multiple processors or for execution on a single processor.
    - A value of 0 means that each query plan is generated so that the query executes on all of the available processors on the database server computer. In general, parallel query execution is not recommended, because of its effect on scalability.
    - A value of 1 means that each query plan is generated so that the query executes on only one processor. In other words, this value turns off parallelism for query execution. Using one processor for query execution is recommended. For this option, in the SQL Server Properties screen select the Processor tab, and in the Parallelism section select Use 1 Processor.

  Also use a single processor for query execution for the component Enterprise Integration Mgr (alias EIM), for Siebel EIM. Do this even when you are using parallel Siebel EIM threads.
- **auto create statistics.** This parameter allows SQL Server to create new statistics for database columns as needed to improve query optimization. Enable this option.
- **auto update statistics.** This parameter allows Microsoft SQL Server to automatically manage database statistics and update them as necessary to promote proper query optimization. Enable this option.

  Turn both auto create statistics and auto update statistics off when running concurrent Siebel EIM threads and performing a full scan of your tables. For information about running full scans, see *Updating Microsoft SQL Server Statistics*.
- **tempdb.** This parameter specifies the database that Microsoft SQL Server uses for the temporary space needed during the execution of various queries. Set the initial size of TEMPDB to a minimum of 100 MB. Also configure the parameter to allow auto-growth, which allows SQL Server to expand the temporary database as needed to accommodate your activity.

Guidelines for Selecting a Language for Microsoft SQL Server

This topic is part of *Configuring a Microsoft SQL Server Database or Azure SQL Database for Siebel CRM*.

As part of database creation, you must set the language characteristics of your database, even if you deploy in only one language.

Sort Order

The sort order is a characteristic that requires special consideration regarding the Siebel database. On Microsoft SQL Server, the sort order of a database instance is specified during database creation and defines the way in which the instance will sort character data.

Although each SQL Server system database and each object within a database can have its own unique sort order, you must set the sort order at the database instance level only.

Siebel CRM support for a given sort order depends both on the code page of the database and on whether it will be used in a development or a production environment.

- **Development environment databases.** The repository object names in your development environment database must sort using binary sort order, because Siebel Tools uses this sort order internally.

  > **CAUTION:** When Microsoft SQL Server is installed, the instance is set by default to dictionary sort order and, if this is not changed, then every database inherits this setting. The master database cannot be changed without rebuilding the instance. Therefore, it is strongly recommended that the instance sort order be set to binary at installation time. Consult your Microsoft SQL Server documentation for instructions on setting this sort order.

  > **Note:** Binary sort order is the simplest and fastest sort order to perform in the database. Binary sorting is case-sensitive and is based on the numeric values (for example, 0 through 255 for an 8-bit character set) of the characters in the installed character set.

  Customers are responsible for making sure that their data is backed up and restored correctly.

- **Production environment databases.** Binary or dictionary sort orders can be used in production environment databases. Binary sort order can give improved performance. See also Microsoft SQL Server documentation. Microsoft recommends that the server level and database-level collations match.

Platform, database, and other certifications for Siebel are found on My Oracle Support. For help searching for certifications, see *How to Check the Certification Matrix in the New My Oracle Support Site* (KB871388).

Related Topics

Planning RDBMS Installation and Configuration

Specifying the Locale for Siebel CRM

Verifying Installation for the Siebel Database

Guidelines for Creating the Microsoft SQL Server Database

This topic is part of *Configuring a Microsoft SQL Server Database or Azure SQL Database for Siebel CRM*.

Use a small, nonproduction environment for testing purposes.

After you install the Database Configuration Utilities on the Siebel Server computer, as described in *Siebel CRM Download and Installation*, you can modify the database table and index creation scripts to specify the tablespace names that you created for Siebel tables and indexes. For more information, see *Guidelines for Overriding Microsoft SQL Server Default Tablespaces for Database Objects*.

For Siebel CRM language support and Unicode support, see the Siebel Global Deployment guide.

Guidelines for Allocating Microsoft SQL Server Database Log Space

This topic is part of *Configuring a Microsoft SQL Server Database or Azure SQL Database for Siebel CRM*.

You must place your log file on a disk large enough to hold the log file as it expands. Monitor the disk regularly for its level of utilization. Very large transactions might, for example, require at least 1 GB.

Guidelines for Overriding Microsoft SQL Server Default Tablespaces for Database Objects

This topic is part of *Configuring a Microsoft SQL Server Database or Azure SQL Database for Siebel CRM*.

Siebel CRM provides the option of overriding the default storage parameters for the tablespaces that you create to hold specific tables or indexes. To do this, edit the ddl.ctl file located in the `DBSRVR_ROOT\MSSQL` directory.

Note: The ddl.ctl file must not be modified except by a qualified DBA.

For each Siebel table, you can specify a tablespace by using the Table Space parameter. In the following example, the tablespace for the table S_APP_VIEW is set to `DATA1`. As provided, the .ctl file does not set storage parameters for the objects it creates, so the objects default to the parameters of the tablespaces in which they are created.

As shown in the example that follows, you can use the Table Space parameter to set storage parameters for specific tables:

```
[Object 219]
Type = Table
Name = S_APP_VIEW
Column 1 = ROW_ID VARCHAR(15) NOTNULL
Column 2 = CREATED TIMESTAMP NOTNULL DEFAULT %NOW%
Column 3 = CREATED_BY VARCHAR(15) NOTNULL
Column 4 = LAST UPD TIMESTAMP NOTNULL DEFAULT %NOW%
Column 5 = LAST_UPD_BY VARCHAR(15) NOTNULL
Column 6 = DCKING_NUM NUMERIC(22,7) DEFAULT 0
Column 7 = MODIFICATION_NUM NUMERIC(10,0) NOTNULL DEFAULT 0
```

```
Column 8 = CONFLICT_ID VARCHAR(15) NOTNULL DEFAULT '0'
Column 9 = NAME VARCHAR(50) NOTNULL
Column 10 = DESC_TEXT VARCHAR(255)
Column 11 = LOCAL_ACCESS_FLG CHAR(1)
Table Space = data1
```

For an example (IBM DB2) of overriding the defaults for specific tables and indexes, see *Guidelines for Overriding IBM DB2 Default Tablespaces for Database Objects*.

Guidelines for Creating Microsoft SQL Server Database Objects

This topic is part of *Configuring a Microsoft SQL Server Database or Azure SQL Database for Siebel CRM*.

Siebel CRM provides sample scripts (CrBlankMSSQLSiebelDEMO.sql and CrBlankMSSQLSiebelDEMO.bat), located in the `DBSRVR_ROOT\MSSQL` directory. Use these scripts as a reference to help you create your own scripts for creating a blank Siebel database based on the requirements for your deployment.

Use the settings in the script as guidelines for your initial configuration. Your final settings will vary depending on the server hardware configuration, the number of users, and the type of workload. Use a small, nonproduction environment for testing purposes.

Before using these scripts, read the file CrBlankMSSQLSiebelDEMOReadMe.txt. See also *Overview of Database Configuration*.

After you install the Database Configuration Utilities on the Siebel Server computer, as described in *Siebel CRM Download and Installation*, you can modify the database table and index creation scripts to specify the tablespace names that you created for Siebel tables and indexes. For more information, see *Guidelines for Overriding Microsoft SQL Server Default Tablespaces for Database Objects*.

Additional information about Microsoft SQL Server configuration and tuning options is available from Microsoft, your hardware vendor, and other sources.

Guidelines for Ongoing Microsoft SQL Server Administration

This topic is part of *Configuring a Microsoft SQL Server Database or Azure SQL Database for Siebel CRM*.

After you have installed Siebel CRM on Microsoft SQL Server, some other tasks must be performed on a periodic basis. These are in addition to such common database administration tasks as monitoring and backing up.

Updating Microsoft SQL Server Statistics

This topic is part of *Configuring a Microsoft SQL Server Database or Azure SQL Database for Siebel CRM*.

The cost-based optimizer in Microsoft SQL Server uses statistics about tables and indexes to compute the most efficient access plans. When the statistics become inaccurate, as can happen for tables with high insertion or deletion rates and for their associated indexes, the performance of database operations can degrade dramatically.

Perform a full scan of all of the tables under the following circumstances, even if you implement automatic statistics updating:

- After installing the Siebel database and before starting Siebel applications.
- After running concurrent Siebel EIM threads.
- After inserting, updating, or deleting large amounts of data.

Using Query Analyzer, perform a full scan of each table by entering the following command:

```
update statistics TableName with full scan
```

It is strongly recommended that you enable the automatic creation and updating of statistics, by using the parameters documented in *Configuring a Microsoft SQL Server Database or Azure SQL Database for Siebel CRM*. This way, statistics are automatically kept up to date and the administrative overhead of updating them manually is removed.

If you do not implement automatic statistics updating, then periodically perform the full scan described in this topic.

Managing Microsoft SQL Server Fragmentation

This topic is part of *Configuring a Microsoft SQL Server Database or Azure SQL Database for Siebel CRM*.

Use the following Microsoft SQL Server command to determine whether a clustered index and its associated tables are highly fragmented:

```
DBCC SHOWCONTIG
```

If this command returns a value for scan density of less than 60%, then use the following Microsoft SQL Server command to defragment tables without having to drop indexes:

```
DBCC INDEXDEFRAG
```

You might want to use this option periodically against the entire database.

If `DBCC SHOWCONTIG` returns a value of less than 30%, or if you suspect that indexes might be interleaved on the disk, then consider rebuilding the index, by using the following command:

```
DBCC DBREINDEX
```

For more information about monitoring fragmentation, see Microsoft SQL Server documentation.

Configuring Siebel Server Connectivity to the Siebel Database

Review this topic to configure connectivity to the Siebel database from the computer where you will install Siebel Server.

The database will already have been created, as described in this chapter.

Oracle Database

Siebel Server connections to the Oracle Database are made through dedicated server processes rather than through Oracle MTS; the use of MTS might negatively affect performance.

Use Oracle's Easy Configuration utility to define a database alias with the proper connection information for your Siebel database. Record the connect string in *Siebel Deployment Planning Worksheet*. You specify this connect string when configuring the Siebel Enterprise.

> **Note:** (UNIX) Siebel CRM supports the Oracle 32-bit client. Therefore, if you have installed the Oracle 64-bit client on your Siebel Server, then you must include `$ORACLE_HOME/lib32` in your definition for the LIBPATH (AIX), SHLIB_PATH (HP-UX), or LD_LIBRARY_PATH (Linux) environment variable. If you require `$ORACLE_HOME/lib` for non-Siebel applications, then make sure that `$ORACLE_HOME/lib32` is placed in front of `$ORACLE_HOME/lib`.

IBM DB2

Define a database alias with the proper connection information for your Siebel database. Record the connect string in *Siebel Deployment Planning Worksheet*. You specify this connect string when configuring the Siebel Enterprise.

You can use the DB2 command line processor to define your database alias. For more information, see IBM DB2 documentation.

IBM DB2 for z/OS

For information about configuring database connectivity for IBM DB2 for z/OS, see *Implementing Siebel Business Applications on DB2 for z/OS* .

Microsoft SQL Server or Azure SQL Database Server

No configuration is required after the Microsoft SQL Server ODBC driver has been installed on each computer.

Siebel CRM automatically creates an ODBC data source using connectivity parameters that you specify when configuring the Siebel Enterprise. Record this data source in *Siebel Deployment Planning Worksheet*.

> **Note:** For Microsoft SQL Server deployments, valid versions of SQL Server Native Client, and Microsoft Data Access Components (MDAC) must reside on the server computer where you will install the Siebel Server.

12 Siebel Deployment Planning Worksheet

Siebel Deployment Planning Worksheet

This chapter provides the Deployment Planning Worksheet. It includes the following topics:

- *About the Deployment Planning Worksheet*
- *Team Lead Summary*
- *Siebel Enterprise Server Names and Installation Directories*
- *Siebel Accounts, Host Names, and Static IP Addresses*
- *Cluster Deployment Data*
- *Ports and RDBMS Details Data*

About the Deployment Planning Worksheet

The topics that follow provide the Deployment Planning Worksheet for installing Oracle's Siebel CRM product family. This worksheet is an integral part of the installation process.

Before proceeding with installation-related tasks, make a copy of the worksheet topics that follow, or create your own worksheet to capture all of the data that you need to record. Record all instances of any modules for which you install multiple instances. For example, Siebel Gateway multiple nodes can participate in a cluster and multiple Siebel Application Interface or Siebel Server instances can provide different Siebel CRM applications or functions or can support load balancing and high availability). The deployment team fills out the first section. Members of the team fill out the information in the sections for which they are responsible. As you work through preparation steps, record the information that you will need while installing and configuring Siebel CRM.

In various places in this book, you are prompted to refer to the Deployment Planning Worksheet for specific information about your site and deployment. You also use it to record other important information for future installations, upgrades, reconfiguration, and expansion of your deployment.

Each time that you install new Siebel CRM software in your deployment, record the relevant data in your Deployment Planning Worksheet. Use a new worksheet for each Siebel CRM environment or Siebel CRM deployment.

> **CAUTION:** Customers are responsible for ensuring the security of sensitive information, such as account passwords, that might be recorded in this worksheet or in similar documents or information stores created by or employed by the customer.

Team Lead Summary

Section 1: Deployment Team Members

Use the following table to record development team members.

Heading	Value (Name)
Deployment Team Lead	*Deployment Team Lead:*
Siebel Administrator	*Siebel Administrator:*
System Administrator	*System Administrator:*
Database Administrator	*Database Administrator:*

Section 2: Deployment Overview

Use the following table to record key overview information for your Siebel CRM deployment.

Component Name	Version	Codepage or Unicode	Owner	Number of Users	Server Operating System
Database Server (RDBMS)	*Version:*	*Codepage:* OR *Unicode:*	*Owner:*	*No. of Users:*	*O/S:*
Siebel Gateway	*Version:*	*Codepage:* OR *Unicode:*	*Owner:*	*No. of Users:*	*O/S:*
Siebel Server	*Version:*	*Codepage:* OR *Unicode:*	*Owner:*	*No. of Users:*	*O/S:*
Siebel Application Interface	*Version:*	*Codepage:* OR *Unicode:*	*Owner:*	*No. of Users:*	*O/S:*

Component Name	Version	Codepage or Unicode	Owner	Number of Users	Server Operating System

Siebel Enterprise Server Names and Installation Directories

Make a copy for *each* Siebel Enterprise Server that you install.

ODBC data sources are created automatically by the Siebel CRM installer. For more information, see *Planning RDBMS Installation and Configuration*.

Section 3: Server Names

Use the following table to record the server names for your Siebel CRM deployment.

Heading	Value
Siebel Enterprise Name	*Siebel Enterprise Name:*
ODBC Data Source Name	*ODBC Data Source Name:*
Primary Language	*Primary Language:*
Other Deployed Languages	*Other Deployed Languages:* **Note:** Different languages might be deployed on different servers. Keep track of all of the languages deployed and the servers on which they are deployed.
Siebel Gateway component	*Network Host Name:* *Installation Directory:*
Siebel Server component	*Network Host Name:* *Installation Directory:*
Database Configuration Utilities component	*Network Host Name:* *Installation Directory:*
Siebel Application Interface component	*Network Host Name:* *Installation Directory:*

Heading	Value
Siebel File System Directories component	*Network Host Name:* *Installation Directory:*

Siebel Accounts, Host Names, and Static IP Addresses

Make a copy for *each* Siebel Enterprise Server that you install.

Note: Requirements vary for user accounts mentioned in this topic. For example, the anonymous user for employee applications must be defined as an employee within the Siebel database. For more information, see *Requirements for Configuring Anonymous Users for the Siebel Application Interface*.

Section 4: Siebel Account Information

Use the following table to record the account information for your Siebel CRM deployment.

Heading	Login or User ID	Password
Siebel Service Owner	*Login:* OR *User ID:*	*Password:*
Siebel Administrator	SADMIN	*Password:*
Employee Anonymous User	*Login:* OR *User ID:*	*Password:*
Additional Anonymous User	*Login:* OR *User ID:*	*Password:*
Security User	*Login:* OR *User ID:*	*Password:*

Section 5: Host Name and Static IP Addresses

Use the following table to record the host names and static IP address information for your Siebel CRM deployment.

Server Name	Static IP or Host Name	Subnet Mask
Siebel Gateway	*Static IP:* OR *Host Name:*	*Subnet Mask:*
Siebel Server	*Static IP:* OR *Host Name:*	*Subnet Mask:*

Cluster Deployment Data

Make a copy for *each* partition of your shared disk (for example: `H:\`, `I:\`, `J:\`, and so on). (Choice of resource groups clustered is optional.)

> **Note:** You can optionally configure Siebel Gateway clustering using Siebel Management Console, as described in *Configuring the Siebel Gateway Cluster*.

Section 6: Cluster Node Network Names and IP Addresses

Use the following table to record the cluster node network names and IP address information for your Siebel CRM deployment.

Heading	Value
Cluster Node Network Name 1	*Network Name 1:*
Cluster Node Network Name 2	*Network Name 2:*
Cluster Node 1 IP Addresses	*IP Address Node 1:*
Cluster Node 2 IP Address	*IP Address Node 2:*
Subnet Mask for All Sample Addresses	*Subnet Mask:*
Physical Disk Name (example: D)	*Disk Name:*

Heading	Value

Section 7: Resource Group Configuration

Use the following table to record the resource group configuration information for your Siebel CRM deployment.

Resource Group Name	Physical Disk Name	IP Addresses	Network Names	Generic Service or File Share
Siebel Gateway Group	*Disk Name:*	*IP Addresses:*	*Network Names:*	*Service:* OR *File Share:*
Siebel File System Group	*Disk Name:*	*IP Addresses:*	*Network Names:*	*Service:* OR *File Share:*
Siebel Server Group	*Disk Name:*	*IP Addresses:*	*Network Names:*	*Service:* OR *File Share:*
Siebel Application Interface Group	*Disk Name:*	*IP Addresses:*	*Network Names:*	*Service:* OR *File Share:*

Ports and RDBMS Details Data

Section 8: Port Numbers

Make a copy for *each* Siebel Enterprise Server that you install.

> **Note:** If your database is IBM DB2 for z/OS, then use the Deployment Planning Worksheet dedicated to the Siebel database installation that is provided in *Implementing Siebel Business Applications on DB2 for z/OS* .

Use the following table to record the port number information for your Siebel CRM deployment.

Siebel Component	Other Ports	HTTPS Redirect Port (Application Container Port)	HTTP Connection Port (Application Container Port)	Shutdown Port (Application Container Port)
Siebel Gateway	*Other Ports:*	*HTTP Redirect Port:*	*HTTP Connection Port:*	*Shutdown Port:*
Siebel Gateway Registry	(Registry Client Port)	Not applicable	Not applicable	Not applicable
Siebel Server	Not applicable	*HTTP Redirect Port:*	*HTTP Connection Port:*	*Shutdown Port:*
Siebel Application Interface	Not applicable	*HTTP Redirect Port:*	*HTTP Connection Port:*	*Shutdown Port:*
Siebel Server: Synchronization Manager (for Siebel Remote)	*Other Ports:*	Not applicable	Not applicable	Not applicable
Siebel Server: Server Request Processor	*Other Ports:*	Not applicable	Not applicable	Not applicable
Siebel Server: Siebel Connection Broker	*Other Ports:*	Not applicable	Not applicable	Not applicable

Section 9: Database Information

Use the following table to record the database information for your Siebel CRM deployment.

Items (if applicable)	Value
RDBMS Operating System, Version and Patch	*RDBMS O/S:* *RDBMS Version:* *RDBMS Patch:*
RDBMS Platform, Version, and Patch	*RDBMS Platform:* *RDBMS Version:* *RDBMS Patch:*
Database Server	*Database Name:*
Host Instance Name	*Host Instance Name:*
Database Name	*Database Name:*

Items (if applicable)	Value
Port Number	*Port Number:*
Table Owner and Password	*Table Owner:* *Table Password:*
Tablespace Name, Usage (Data or Index), and Page Size	*Tablespace Name:* *Tablespace Usage:* *Tablespace Page Size:*
Tablespace Name, Usage (Data or Index), and Page Size	*Tablespace Name:* *Tablespace Usage:* *Tablespace Page Size:*
Tablespace Name, Usage (Data or Index), and Page Size	*Tablespace Name:* *Tablespace Usage:* *Tablespace Page Size:*

13 Siebel Open Integration

Siebel Open Integration

This chapter provides an overview of how Siebel CRM enables external access to business processes through Open Integration. It explains how both standard and customized Siebel components can be exposed as REST APIs for integration with on-premises or cloud applications.

You can install and deploy Siebel Open Integration on Windows and Linux in the cloud or on premises. You can also use Siebel Cloud Manager (SCM) to deploy Siebel Open Integration on a Kubernetes Cluster.

The chapter covers the following topics:

- *Installing Siebel Open Integration*
- *Configuring and Deploying Siebel Open Integration*
- *Design Time Workflow and Supporting Artifacts*
- *Managing Security and Authentication*
- *Configuring Open Integration Enterprise Cache for Siebel*
- *Siebel AI Connectors*
- *Configuring Open Integration Logging*

Installing Siebel Open Integration

This topic describes the steps to install and deploy Siebel Open Integration on Windows and Linux in the cloud or on premises.

For on-premises deployments, you can deploy Siebel Open Integration on Windows or Linux. Choose the platform that matches your deployment requirements.

You can also deploy Siebel Open Integration on a Kubernetes clusters using Siebel Cloud Manager (SCM). For more information, refer to *Deploying Open Integration on a Kubernetes Cluster Using Siebel Cloud Manager Guide*.

Based on your deployment choice, download the installer artifacts, and follow the step-by-step instructions for your deployment type.

For more informtion, see the following:

- *Installing Siebel Open Integration on Windows*
- *Installing Siebel Open Integration on Linux*

Installing Siebel Open Integration on Windows

This topic describes how to deploy Siebel Open Integration on Windows in interactive mode and silent mode. It includes the following sections:

- *Installing Siebel Open Integration on Windows in Interactive Mode*
- *Installing Siebel Open Integration on Windows in Silent Mode*

Installing Siebel Open Integration on Windows in Interactive Mode

After you download the Windows installer artifacts for Siebel Open Integration, the following folders and files are included: `archives, ext, jre, resources, openint_installer.properties, README, runInstaller (bat).` . You must read the instructions in the `README` before starting the installation.

To install Siebel Open Integration on Windows using the installer in interactive mode, do the following:

1. Run `runInstaller.bat` to start the GUI installer.
2. Select the installation directory, and then click **Next**.
3. On the Component Selection screen, select the Siebel Open Integration checkbox, and then click **Next**.
4. On the Summary screen, review the installation information, and then click **Install**. When the installation completes, the progress screen displays Installation Completed.
5. Click **Close** to exit the installer.

After the installation completes, go to the installation directory that you specified during installation to view the Siebel Open Integration artifacts and folders that were created. The installed directory includes the following folders and files:

```
apache-maven, compile_dependency, graalvm-jdk, logs, openint-deploy, uninstall, jq
(app), jq (file), siebelinstall.rsp, start (shell), settings.xml.template, start (bat)
```

.

Installing Siebel Open Integration on Windows in Silent Mode

To perform a silent installation of Siebel Open Integration:

1. Generate a response file by launching the installer and proceed to the Summary screen.
2. On the Summary screen, click **Save Response File**, and then save the file as `<Response File Name>.rsp` in the directory of your choice.
3. Run the following command to install Siebel Open Integration in silent mode:

   ```
   runInstaller.bat -silent -responseFile <Response File Name>.rsp
   ```

   In the above command, include the full path of the response file.

   After the command runs successfully, Siebel Open Integration is installed in the installation directory that you specified.

> **Note:** Verify that the `.rsp` file specifies a valid path for `SIEBEL_HOME_LOCATION`. If this directory was used for a previous GUI installation, edit the `.rsp` file to point to a new, empty directory. Make sure the target directory is empty and that the user has read and write permissions.

Installing Siebel Open Integration on Linux

This topic describes how to deploy Siebel Open Integration on Linux in interactive mode and silent mode. It includes the following sections:

- *Installing Siebel Open Integration on Linux in Interactive Mode*
- *Installing Siebel Open Integration on Linux in Silent Mode*

Installing Siebel Open Integration on Linux in Interactive Mode

Use a VNC session for the Linux GUI installation. After you download the Siebel Open Integration Linux installer artifacts, the following folders and files are included: `archives`, `ext`, `jre`, `resources`, `openint_installer.properties`, `README`, `runInstaller (sh)`. Before starting the installation, review the instructions in the README.

To install Siebel Open Integration on Linux using the installer in interactive mode, do the following:

1. Run `runInstaller.sh` to start the GUI installer.
2. Select the installation directory, and then click **Next**.
3. On the Component Selection screen, select the Siebel Open Integration checkbox, and then click **Next**.
4. On the Summary screen, review the installation information, and then click **Install**. When the installation completes, the progress screen displays **Installation Completed**.
5. Click **Close**to exit the installer.

After the installation completes, go to the installation directory that you specified during installation to view the Siebel Open Integration artifacts and folders that were created. The installed directory includes the following folders and files:

```
apache-maven, compile_dependency, graalvm-jdk, logs, openint-deploy, uninstall, jq
(app), jq (file), siebelinstall.rsp, start (shell), settings.xml.template, start (bat).
```

Note: Make sure that the installation directory is empty and that the Linux user has read and write permissions for it.

Installing Siebel Open Integration on Linux in Silent Mode

To perform a silent installation of Siebel Open Integration:

1. Generate a response file by launching the installer and proceed to the **Summary** screen.
2. On the **Summary** screen, click **Save** Response File, and then save the file as `<Response File Name>.rsp` in the directory of your choice.
3. Run the following command to install Siebel Open Integration in silent mode:

   ```
   runInstaller.sh -silent -responseFile <Response File Name>.rsp
   ```

   In the above command, include the full path of the response file.

   After the command runs successfully, Siebel Open Integration is installed in the installation directory that you specified.

Note: Verify that the `.rsp` file specifies a valid path for `SIEBEL_HOME_LOCATION`. If this directory was used for a previous GUI installation, edit the `.rsp` file to point to a new, empty directory. Make sure the target directory is empty and that the user has read and write permissions for the Linux user.

Configuring and Deploying Siebel Open Integration

This topic describes how to configure and deploy Siebel Open Integration. It includes the following sections:

- *Prerequisites for Configuring and Deploying Siebel Open Integration*
- *Configuring Siebel Open Integration*
- *Deploying Siebel Open Integration*

Prerequisites for Configuring and Deploying Siebel Open Integration

You must perform the following tasks before configuring and deploying Siebel Open Integration:

1. Install Powershell for Windows: If you run Siebel Open Integration on Windows, you must install PowerShell. The pipeline requires PowerShell to run.

   > **Note:** After installing Powershell, you must unblock the pipeline helper files, as follows:
   > - Click **describe.ps1** and **fndrep.ps1.**
   > - Select **Properties**.
   > - Select **Unblock.**
   > - Click **OK** or **Apply**.
   >
   > For more information, refer *Install PowerShell 7 on Windows*.

2. Install `wget` on Linux: On Linux, the pipeline uses the wget command to download the OpenAPI specification from the configured Siebel CRM environment.
3. (Optional) Deploy Coherence through Siebel Management Console (SMC). This is required only if you plan to use Coherence through SMC for Open Integration user session caching.
4. Configure Siebel Single Sign-On (SSO): You must configure Siebel SSO before you start the Open Integration configuration. Open Integration is designed to work only when SSO is enabled for Siebel Object Managers.

Configuring Siebel Open Integration

This section describes how to configure Siebel Open Integration.

To configure Siebel Open Integration, you must perform the following tasks:

1. (Optional) Configure Git repository to enable versioning, collaboration, and change tracking. For more information, see *Configuring Git Repository*.
2. Prepare Integration Configurations. For more information, see *Preparing Integration Configurations*.
3. Configure the Siebel Server connection details and the Siebel CRM artifacts to expose through REST APIs in config.json. For more information, see *Configuring config.json*.

4. Define Siebel Open Integration security settings and the credentials required to communicate with Siebel CRM inprofile.json. For more information, see *Configuring profile.json*.
5. Configure the Maven settings in settings.xml. For more information, see *Configuring Maven Settings*.
6. Run the design-time pipeline. For more information, see *Running the Design Time Pipeline*.

Configuring Git Repository

You can connect a Git repository when you manage extensibility artifacts such as configurations, scripts, and Domain-Specific Languages (DSLs), in source control. Connecting a Git repository enables version control, collaboration, and change tracking for these artifacts.

After you configure the Git repository, you can commit updates through your standard Git workflow, including branching, pull requests, and reviews.

To configure a Git repository:

1. Create `git.json` by cloning `git_template.json`.
2. Configure `git.json`.

Configuring git.json

The `git.json` file defines the remote Git repository, branch, and authentication settings used by the following scripts:

- `gen.sh` or `gen.bat` with `--download-from-git`
- `gitUpload.sh or gitUpload.bat`

You can configure the following parameters in `git.json`:

Parameter	Used in	Description
uploadTargetBranch	Upload	(Optional) Use this parameter to specify the branch to push HEAD to. If blank, defaults to `uploadCheckoutBranch` .
uploadCheckoutBranch	Upload	(Optional) Use this parameter to specify the branch to check out or create locally before upload. If blank, defaults to main.
sshPrivateKeyPath	Download and upload	(Optional) Use this parameter to specify the private key file path used only for SSH URLs. If blank, the default SSH configuration or agent is used. The key can be placed in the current working directory, or you can provide an absolute path.
remoteRepositoryUrl	Download and upload	(Required) Use this parameter to specify the remote Git repository URL to clone, push, or tag against - HTTP, HTTPS, or SSH.
integrationConfigurationsDirectory	Download and upload	(Optional) Relative path inside the cloned repository used for integration configurations.`integrationConfigurationsDirecto`r acts as the cloned content root for the Git sync payload. Under this root, the following content is read or written:

Parameter	Used in	Description
		• `integration-configurations/` • `config.json` • `profile.json` In the download flow, content is copied from this location to the local workspace. In the upload flow, content is copied from the local workspace to this content root. The upload flow creates the content root when it is missing. If `integrationConfigurationsDirectory` is omitted or blank, the cloned repository root is used as the default cloned content root.
httpsTrustSourcePath	Download and upload	(Optional) Use this parameter to specify the HTTPS trust source path. This parameter accepts the following inputs: • Certificate files, such as `.pem`, `.crt`, or `.cer`, or • Truststores, such as .jks, .p12, or .pfx. For some JKS files, Java can still load certificates with a null password. In this case, the integrity check is skipped, and Git operations may still succeed. Certificates or truststores can be placed in the current working directory, or you can provide an absolute path. This field is ignored for SSH repository URLs.
gitUsername	Download and upload	(Optional) Use this parameter to specify the user name for HTTP or HTTPS authentication. Used with GIT_TOKEN for authenticated access.
downloadSourceBranch	Download	(Required) Use this parameter to specify the branch used as the clone source for download.

Note:

- For HTTP or HTTPS repository URLs, if you provide `gitUsername`, you must also set `GIT_TOKEN` by using `--git-token` in the entry scripts (`gen.sh` or `gen.bat`).
- For truststore inputs in `httpsTrustSourcePath`, pass the truststore password in the entry scripts using one of the following parameters:
  - `--git-trusttore-password`
  - `--git-truststore-password`

This is strongly recommended for truststore files. For some JKS files, Java can still load certificates with a null password. In this case, the integrity check is skipped, and Git operations may still succeed.

Sample git.json

```
{
 "remoteRepositoryUrl": "git@github.com:openint-conf/siebel-open-integration-configurations.git",
 "downloadSourceBranch": "main",
 "gitUsername": "",
 "uploadCheckoutBranch": "feature/openint-config-sync",
 "uploadTargetBranch": "main",
 "sshPrivateKeyPath": "/Users/username/.ssh/id_rsa",
 "integrationConfigurationsDirectory": "integration-configurations",
 "httpsTrustSourcePath": "/Users/username/certs/internal-ca.pem"
}
```

Preparing Integration Configurations

Integration configurations define how external requests are processed end to end when they interact with Siebel CRM. They provide a declarative way to compose and govern integrations by combining the following capabilities:

- Orchestration through DSL-based step sequencing and control flow.
- Data transformation mappings.
- Resilience features, such as retries and error handling.
- Event-driven publish and subscribe patterns.

Integration configurations help you to:

- Standardize how integrations run.
- Control success and failure paths.
- Change integration behavior with minimal custom code.

This approach supports scalable, extensible, and maintainable connectivity between Siebel and other enterprise applications.

Before you configure an integration, identify the business process that you want to automate. For example, an external application might submit a service request, Siebel CRM might validate and process the request, and another enterprise system might need to be notified when processing is complete.

Integration configurations define this end-to-end process, including how requests enter Siebel CRM, how data is transformed, which actions are executed, how errors are handled, and how external systems participate in the integration flow.

To prepare integration configurations:

1. Prepare JSONata configurations.
2. Prepare Domain Specific Languages (DSLs).
3. Prepare domain configurations.

> **Note:** Open Integration transformation only supports integrations that use Siebel APIs or external application APIs that communicate with Siebel CRM and require access to Siebel metadata for transformation processing.

Preparing JSONata Configurations

In many integrations, the payload received from an external system does not match the format required by Siebel CRM. JSONata mappings enable you to transform incoming and outgoing payloads by using declarative expressions instead of custom code.For example, an external application might use different field names or data structures than those required by Siebel CRM.

JSONata provides a concise, declarative way to select, filter, combine, and transform JSON data. JSONata mappings can convert the payload into the required format before processing and transform Siebel CRM responses into the format expected by the external application.This helps integrations map between external API contracts and the structures expected by Siebel processes, and from Siebel processes back to external API contracts, without custom procedural code.

Transformation mappings in integration configurations use the JSONata expression language to define how request and response payloads are reshaped. By embedding JSONata-based mappings in the configuration, you can implement and change payload transformations in a controlled and maintainable way.

This approach helps keep integration logic consistent and easier to test and support.

Sample JSONata Configuration

```
(
 $items := $.get_product_order.SiebelMessage.ListOfSWIOrderIO.SWIOrder[0].ListOfSWIOrderItem.SWIOrderItem;

 $safeNumber := function($val) {
 $trim($string($val)) != "" ? $number($val) : 0
 };

 $buildTree := function($nodes, $parentId, $headerPriceListId) {
 $map(
 $filter($nodes, function($n) {
 $n.'Parent Order Item Id' = $parentId
 }),
 function($node) {
 (
 $children := $buildTree($nodes, $node.Id);

 $purchaseChild := $filter($nodes, function($c) {
 $c.'Parent Order Item Id' = $node.Id and
 $contains($c.Name, "-PURCHASE")
 })[0];

 $purchasePriceToAdd := $purchaseChild ? $safeNumber($purchaseChild.'Net Price') : 0;

 $netPrice := $safeNumber($node.'Net Price');
 $listPrice := $safeNumber($node.'List Price');

 $hasDiscountPercent :=
 $exists($node.'Discount Percent') and
 $trim($string($node.'Discount Percent')) != '';

 $hasDiscountAmount :=
 $exists($node.'Discount Amount') and
 $trim($string($node.'Discount Amount')) != '';

 $hasUnitPrice :=
 $exists($node.'Unit Price') and
 $trim($string($node.'Unit Price')) != '';

 $hasPriceListId :=
 $exists($node.'Price List Id') and
 $trim($string($node.'Price List Id')) != '';

 $resolvedPriceType :=
 $hasDiscountPercent ? 'discountPercentageOverride' :
 $hasDiscountAmount ? 'amountOverride' :
 $hasUnitPrice ? 'discountOverride' :
 null;

 $priceListId :=
```

```
$hasPriceListId ? $node.'Price List Id' : $headerPriceListId;

$base := {
'@type': 'ProductOrderItemOSMDBE',
'id': $node.Id,
'action':
$lowercase($node.'Action Code') = 'new'
? 'add'
: $lowercase($node.'Action Code'),

'quantity': $node.Quantity,

'productOffering': {
'id': $node.'External Product Id',
'name': $node.'Name'
},

'priceList': {
'id': $priceListId,
'name': ''
},

'itemPrice': [
$merge([
{
'priceType': $node.'Price Type',
'recurringChargePeriod': $node.'Unit of Measure',
'price': {
'dutyFreeAmount': {
'value': $listPrice,
'unit': $node.'Currency Code'
},
'taxIncludedAmount': {
'value': $listPrice,
'unit': $node.'Currency Code'
}
}
},
$resolvedPriceType
? {
'priceAlteration': [
{
'priceType': $resolvedPriceType,
'price': {
'dutyFreeAmount': {
'unit': $node.'Currency Code',
'value': $netPrice
},
'taxIncludedAmount': {
'unit': $node.'Currency Code',
'value': $netPrice
}
}
}
]
}
: {}
])
],

'itemTotalPrice': (
$promoType := $node.'Promotion Role';
$mrc := $safeNumber($node.'Promotion MRC Total');
$nrc := $safeNumber($node.'Promotion NRC Total');

$promoType = 'Bundled Promotions'
```

```
? [
{
'priceType': 'recurring',
'price': {
'dutyFreeAmount': {
'value': $mrc,
'unit': $node.'Currency Code'
},
'taxIncludedAmount': {
'value': $mrc,
'unit': $node.'Currency Code'
}
}
},
{
'priceType': 'oneTime',
'price': {
'dutyFreeAmount': {
'value': $nrc,
'unit': $node.'Currency Code'
},
'taxIncludedAmount': {
'value': $nrc,
'unit': $node.'Currency Code'
}
}
}
]
: []
),

'product': {
'@type': 'ProductRefOrValueOSMDBE',
'id': $node.'Asset Integration Id',
'serviceId': $node.'Service Id',
'name': 'emulator',

'productSpecification': {
'id': $node.'External Class Id',
'name': $node.'Class Name'
},

'productCharacteristic': (
$xa := $node.ListOfSWIOrderItemXA.SWIOrderItemXA;

$map(
$type($xa) = "array"
? $xa
: $type($xa) = "object"
? [$xa]
: [],
function($c) {
{
'name': $c.Name,
'value': $c.Value,
'valueType': $c.'Property Type Code'
}
}
)
)
}
};

$filteredChildren := $filter($children, function($v) {
$not($contains($v.productOffering.name, "-PURCHASE"))
});
```

```
$count($filteredChildren) > 0
? $merge([
$base,
{
'productOrderItem': $filteredChildren
}
])
: $base
)
}
)
};

{
"@type": "ProductOrderOSM_Create",

'externalId':
$.get_product_order.SiebelMessage.ListOfSWIOrderIO.SWIOrder[0].Id,

'description':
$.get_product_order.SiebelMessage.ListOfSWIOrderIO.SWIOrder[0].Description,

'priority':
$.get_product_order.SiebelMessage.ListOfSWIOrderIO.SWIOrder[0].'Order Priority',

'orderTotalPrice': [
{
'priceType': 'recurring',
'price': {
'dutyFreeAmount': {
'value': $safeNumber(
$.get_product_order.SiebelMessage.ListOfSWIOrderIO.SWIOrder[0].'MRC Total'
),
'unit':
$.get_product_order.SiebelMessage.ListOfSWIOrderIO.SWIOrder[0].'Currency Code'
},
'taxIncludedAmount': {
'value':
$safeNumber(
$.get_product_order.SiebelMessage.ListOfSWIOrderIO.SWIOrder[0].'Tax Amount MRC'
) +
$safeNumber(
$.get_product_order.SiebelMessage.ListOfSWIOrderIO.SWIOrder[0].'MRC Total'
),
'unit':
$.get_product_order.SiebelMessage.ListOfSWIOrderIO.SWIOrder[0].'Currency Code'
}
}
},
{
'priceType': 'oneTime',
'price': {
'dutyFreeAmount': {
'value': $safeNumber(
$.get_product_order.SiebelMessage.ListOfSWIOrderIO.SWIOrder[0].'NRC Total'
),
'unit':
$.get_product_order.SiebelMessage.ListOfSWIOrderIO.SWIOrder[0].'Currency Code'
},
'taxIncludedAmount': {
'value':
$safeNumber(
$.get_product_order.SiebelMessage.ListOfSWIOrderIO.SWIOrder[0].'Tax Amount NRC'
) +
$safeNumber(
```

```
$.get_product_order.SiebelMessage.ListOfSWIOrderIO.SWIOrder[0].'NRC Total'
),
'unit':
$.get_product_order.SiebelMessage.ListOfSWIOrderIO.SWIOrder[0].'Currency Code'
}
}
}
],

'billingAccount': {
'id':
$.get_product_order.SiebelMessage.ListOfSWIOrderIO.SWIOrder[0].'Billing Account Id',
'name':
$.get_product_order.SiebelMessage.ListOfSWIOrderIO.SWIOrder[0].'Billing Account'
},

'productOrderItem': [
$buildTree(
$items,
"",
$.get_product_order.SiebelMessage.ListOfSWIOrderIO.SWIOrder[0].'Price List Id'
)
]
}
)
```

For information about JSONata, refer the *JSONata documentation*.

Preparing DSLs

After you define the required data transformations, define the integration execution flow by using a domain-specific language (DSL). A domain-specific language (DSL) defines the runtime behavior of an integration as a configuration-driven workflow. You can use a DSL to model an integration as a sequence of executable steps that coordinate multiple actions, such as invoking Siebel CRM operations, applying transformations, and publishing or consuming events.

DSL provides built-in control flow constructs that define how the integration proceeds under different outcomes. These constructs include:

- Step sequencing to control execution order.
- Success and failure transitions to route processing based on outcomes.
- Conditional branching to support different paths for different scenarios.
- Loops and iteration to iterate over a specific step or an entire workflow as needed.
- Retries and error-handling behavior to improve resilience and standardize responses.

A valid DSL includes the following top-level keys:

- Required top-level keys:
  - `input`
  - `output`
  - `steps`
  - `response`
  - `transitions`
- Optional top-level key is `global`.

Input and Output

You can use `input` and `output` to define the OpenAPI contract for `/integration/{integrationId}`:

- `input` defines the OpenAPI request body schema.
- `output` defines the OpenAPI `200` response schema.
- Step execution uses `steps`, `inputSpecification`, and `response`.

The following rules apply:

- Top-level `input` and `output` are required.
- Both values must be JSON objects.
- Both values must include top-level `"type": "object"`.
- Describe generation fails if `"type": "object"` is missing.

The following schema nodes are supported in `input` and `output` keys:

Parameter	Description
Inline object	Uses `type: "object"` and recursively processes `properties`. Additional schema attributes (for example, `description`, `required`, `additionalProperties`, and `example`) are preserved.
Inline array	Uses `type: "array"` and `items`. Additional array attributes (for example, `minItems`, `maxItems`, `uniqueItems`) are preserved.
Inline primitive schema	Copies primitive schemas, such as `string`, `integer`, `number`, and `boolean`, to OpenAPI as-is. Attributes such as `format`, `enum`, `default`, `minimum`, `maximum`, `nullable` are preserved.
Siebel IO reference	Uses `type: "IntegrationObject"` and `IntegrationObject: "<IO Name>"`. The value is replaced with a `$ref` to a generated component schema resolved from Siebel metadata.
External OpenAPI reference	Uses `type: "externalSchema"`, `restDef`, and $ref. The schema is imported from the configured external REST OpenAPI and rewritten to a local `#/components/schemas/...` reference. For `externalSchema`, the following rules apply: • `restDef` is required and must match `resources.externalREST[].spokesystem` in the integration config. • `$ref` is required and must target a concrete schema. For example, `#/components/schemas/OrderRequest`, `/components/schemas/OrderRequest`. • `$ref` values `#`, `/`, and `#/` are invalid. • When multiple `restDef` systems are used in one DSL, imported component names are namespaced to avoid collisions.

Examples:

- Inline request and response schemas:

```
"input": {
 "type": "object",
 "required": ["customerId", "items"],
 "properties": {
 "customerId": { "type": "string" },
 "priority": { "type": "string", "enum": ["LOW", "NORMAL", "HIGH"], "default": "NORMAL" },
 "items": {
 "type": "array",
 "minItems": 1,
```

```
 "items": {
 "type": "object",
 "required": ["sku", "qty"],
 "properties": {
 "sku": { "type": "string" },
 "qty": { "type": "integer", "minimum": 1 }
 }
 }
 }
 }
},
"output": {
 "type": "object",
 "properties": {
 "orderId": { "type": "string" },
 "status": { "type": "string" },
 "acceptedAt": { "type": "string", "format": "date-time", "nullable": true }
 }
}
```

- `IntegrationObject` Usage:

```
"input": {
 "type": "object",
 "properties": {
 "account": {
 "type": "IntegrationObject",
 "IntegrationObject": "Account"
 }
 }
},
"output": {
 "type": "object",
 "properties": {
 "contacts": {
 "type": "array",
 "items": {
 "type": "IntegrationObject",
 "IntegrationObject": "Contact"
 }
 }
 }
}
```

- `externalSchema` Usage:

```
"input": {
 "type": "object",
 "properties": {
 "quoteRequest": {
 "type": "externalSchema",
 "restDef": "pricing-rest",
 "$ref": "#/components/schemas/QuoteRequest"
 }
 }
},
"output": {
 "type": "object",
 "properties": {
 "quoteResponse": {
 "type": "externalSchema",
 "restDef": "pricing-rest",
 "$ref": "#/components/schemas/QuoteResponse"
```

```
  }
  }
}
```

Step Types and Fields

You can use `steps` to define the executable workflow steps. Supported step types are:

- `BusinessService`
- `Workflow`
- `RESTOutbound`
- `EventPublish`

Note: The `RestOutbound` step type is supported only for the out-of-the-box integrations delivered with Open Integration. Using this step type in custom integrations is not supported.

The following table lists the fields supported for each step type and indicates whether a field is required for Business Service, Workflow, REST Outbound, and Event Publish step types.

Field	BusinessService	Workflow	RESTOutbound	EventPublish	Description
`type`	Required	Required	Required	Required	Step execution type.
`service`	Required	No	No	No	Siebel business service name.
`method`	Required	No	Required	No	Business service method or HTTP method.
`serviceName`	No	Required	No	No	Siebel workflow process.
`restSpec`	No	No	Required	No	External REST alias.
`endPoint`	No	No	Required	No	Relative REST path.
`pathParams`	No	No	Optional	No	Path token values.
`queryParams`	No	No	Optional	No	Query string values.
`inputHeaders`	No	No	Optional	Optional	Request or event headers.
`queue`	No	No	No	Required	Outbound event alias.
`partitions`	No	No	No	Optional	Fixed partitions.
`partitionKey`	No	No	No	Optional	Dynamic partition key.
`inputSpecificatior`	Required	Required	Required	Required	Data assembly pipeline.
`retryConfig`	Optional	Optional	Optional	Optional	Retry behavior.
`output`	Optional	Optional	Optional	Optional	Updates global values from step output.

Field	BusinessService	Workflow	RESTOutbound	EventPublish	Description
`storeInputToGlobal`	Optional	Optional	Optional	Optional	Updates global values from step input.

Data Collection

You can use `dataCollection` in `inputSpecification` to collect data for a step or response. Each entry requires `source`. The optional field is `transformationSpec`.

The supported `source` values are:

Source	Runtime Behavior
`WorkflowInput`	Collects the original inbound payload. Applies source-level `transformationSpec` if provided.
`WorkflowHeaders`	Collects the inbound request headers map. Applies source-level `transformationSpec` if provided.
`<StepName>`	Collects the step output for the named step. Before source-level transformation, runtime augments the input with `global`. On the error path, runtime also augments the input with `exception`.

The final collected object contains one key for each source name. Runtime also adds root-level `global`. In the error path, runtime also adds root-level `exception`.

Important behavior:

- Source processing is parallel, so ordering is not guaranteed.
- If a referenced step has not produced output, source data is absent for that key.
- If source-level `transformationSpec` is not set, transformation is skipped for that source.
- Transformation specs can access global storage without additional settings.

Transformation Specifications

You can use `transformationSpec` to define transformation behavior. It can appear in:

- `dataCollection[].transformationSpec`
- `transformation[].transformationSpec`
- Response payload transformation blocks

The transformation pipeline runs in this order:

1. Collect data from `dataCollection`.
2. Run `transformation`[] sequentially.
3. Run optional script hooks:
   - `preTransformationScript`
   - `postTransformationScript`
4. Run optional `transformationScript` after the transformation chain.

Transformation specification files look up behaviour:

- Runtime reads transformation specs from `integration-configurations/<domain>/jsonata-configurations/transformation-spec/`.

- Only valid listed domains from the merged config are considered.
- Relative subpaths, such as `folder/file.json`, are supported.
- Spec content is cached in memory after the first read.

You must use unique file names across domains to avoid ambiguous first-match behavior. Keep transformations idempotent where possible for retry safety.

Embedded Expressions and Conditions

Open Integration evaluates conditions and expressions with Apache JEXL.

Expression fields use this wrapper `${...}`. Runtime removes the wrapper and evaluates the expression inside it. If a value does not use `${...}`, the direct provided value is used.

You can configure embedded expressions in:

- Transition conditions
- `output` and `storeInputToGlobal` mapping expressions
- REST or Event header, path, and query dynamic values
- Event publish `partitionKey`
- Retry `retryOnCondition`
- Response `statusCode` expressions and header expressions

You must use null-safe expressions and keep expressions free of side effects, and use `global.*` as the stable cross-step state interface.

Open Integration evaluates JEXL expressions in a restricted sandbox. The sandbox allows read-only expression logic and blocks operations that can mutate data or access unsafe Java runtime capabilities.

Allowed Expressions

You can use the following expression types:

- Arithmetic expressions
- Boolean expressions
- `var`
- `for`
- Literals
- Object, map, and list construction syntax

You can read values from:

- `WorkflowInput`
- `WorkflowHeaders`
- `global`
- Prior step contexts

Allowed Data Access

You can read data by using:

- Map or property access, such as `WorkflowInput.foo`
- Quoted keys, such as `ratingList.'Special Rating List Id'`
- Bracket access, such as `map['key']`

Allowed Methods

You can use read-only map and list helper methods.

Allowed map methods include:

- `Map.get`
- `getOrDefault`
- `containsKey`
- `size`

Allowed list methods include:

- `List.get`
- `subList`
- `contains`
- `indexOf`
- `size`

You can also use common safe string methods, such as:

- `trim`
- `substring`
- `toUpperCase`
- `replace`
- `matches`
- `isBlank`

Restricted Operations

You must not use methods that mutate lists or maps. The following operations will fail:

- `result.add(...)`
- `map.put(...)`
- `remove(...)`

The sandbox also blocks Java runtime escape mechanisms, including:

- Class loading
- Reflection

- `Runtime`
- `ProcessBuilder`
- `System`
- Threads

WorkflowInput and WorkflowHeaders

`WorkflowInput` and `WorkflowHeaders` are workflow-level special variables in embedded expressions:

- `WorkflowInput` contains the original inbound request payload for the integration API. You must treat it as read-only and use it when you need stable access to original request fields, independent of transformed step `input`.
- `WorkflowHeaders` contains inbound HTTP headers for the integration API request. You must treat it as a read-only map of header names to values, and use it for correlation, channel, auth-context propagation, and response or header enrichment.

To access the workflow-level special variables:

- Use dot notation for standard keys `WorkflowInput.accountId`.
- Use bracket notation for keys with special characters or spaces, or for explicit header lookup `WorkflowHeaders['x-correlation-id']`.

When defining expressions, choose the variable source based on the data that you need:

- Use `WorkflowInput` and `WorkflowHeaders` to access the original request payload and request headers. These variables provide the initial context that was supplied when the workflow started.
- Use `input` when you need to reference the current step input. This is particularly useful after data collection or transformation steps that modify the data before it is passed to the next step.
- Use `global` to access values that are computed or stored during workflow execution and need to be shared across multiple steps.

Transition conditions

Transition conditions control how the workflow moves from one state to another. The workflow evaluates transitions in the configured order and takes the first matching transition.

After a condition matches, the workflow invokes the target step and then continues into the target state.

Transition conditions typically read state produced by previously executed steps. This state is often stored in `global`.

You must use embedded expressions for transition conditions and dynamic values.

Embedded Expression Examples

- Transition based on `global`:

  ```
  "condition": "${global.account_status == 'Active'}"
  ```

- Transition based on workflow input:

  ```
  "condition": "${WorkflowInput.promotionId != null && WorkflowInput.promotionId != ''}"
  ```

- REST output global mapping by using `status` and `output`:

  ```
  "output": {
   "global.rest_status": "${status}",
  ```

```
 "global.external_ref": "${status == 200 ? output.externalReference : null}"
}
```

- Dynamic header from input and `global`:

```
"inputHeaders": {
 "account-id": "${global.account_id}",
 "channel": "${WorkflowHeaders['x-channel']}"
}
```

- Retry condition:

```
"retryOnCondition": "${output.errorCode != null && output.errorCode != ''}"
```

Embedded Expression Authoring Rules

When you author embedded expressions, follow these rules:

- Use `global.*` as the stable cross-step state interface.
- Update `global` through step output mappings before you use those values in later transition conditions.
- Handle nullable values defensively, such as `x != null && ...`.
- For keys with spaces or special characters, use quoted or bracket access patterns.
- Keep expressions free of side effects.

output and storeInputToGlobal

You can use `output` and `storeInputToGlobal` as global update maps:

- Use the following key format:

  `global.<name>`

- Use the following value format:

  `${...}`

  The value must be an embedded expression.

The following validation rules apply:

- The key must start with `global`.
- `<name>` must exist in the top-level `global` list.
- The value must satisfy the embedded expression format.

retryConfig

You can use `retryConfig` to control retry eligibility, delay strategy, and retry stop conditions.

The following table lists the fields available in the `retryConfig` section:

Field	Type	Description
`retryMaxAttempts`	Integer	Total number of attempts, including the initial call. For example, `5` means the initial attempt plus up to 4 retries.

Field	Type	Description
`retryOnlyOnStatusCodes`	Integer array	HTTP status codes for which retry is attempted. For example, `[502, 503, 504]`.
`retryExcludeStatusCodes`	Integer array	HTTP status codes for which retry is skipped. For example, `[400]`.
`retryOnCondition`	String embedded expression	Expression that is evaluated at runtime after an attempt. If the expression evaluates to `true`, retry is attempted.
`retryDelayMilliseconds`	Integer	Base delay in milliseconds before retry. For example, `1000` is a 1-second base delay.
`retryBackoffMultiplier`	Number	Scales the delay across attempts by using exponential or factor backoff. If set to `1.0`, the delay stays fixed.
`retryMaxDelayMilliseconds`	Integer	Upper limit for the computed retry delay.
`retryJitterMilliseconds`	Integer	Random plus or minus jitter applied to the computed delay to help avoid synchronized retries.
`retryTotalTimeoutMilliseconds`	Integer	Maximum total elapsed time allowed for all attempts and waits.

Delay Progression Model

The retry delay typically uses this progression model: `delay = retryDelayMilliseconds * (retryBackoffMultiplier ^ (attempt_number - 1))`

After the delay is calculated, the runtime applies the following rules:

- Bounds the delay by `retryMaxDelayMilliseconds`, if configured.
- Applies `retryJitterMilliseconds`, if configured.

Validation and Compatibility Rules

The following schema-level constraints apply:

- One of `retryOnlyOnStatusCodes`, `retryExcludeStatusCodes`, or `retryOnCondition` is required.
- One of `retryMaxAttempts` or `retryTotalTimeoutMilliseconds` is required.
- `retryDelayMilliseconds` is required.

Custom Validator Constraints

Step Type	Retry Rules
`BusinessService`	Must use `retryOnCondition`. Must not use `retryOnlyOnStatusCodes` or `retryExcludeStatusCodes`.
`Workflow`	Must use `retryOnCondition`. Must not use `retryOnlyOnStatusCodes` or `retryExcludeStatusCodes`.
`EventPublish`	Must not use `retryOnCondition`, `retryOnlyOnStatusCodes`, or `retryExcludeStatusCodes`.

Step Type	Retry Rules
`RESTOutbound`	Can use expression-based retry with `retryOnCondition`, or status-code-based retry with `retryOnlyOnStatusCodes` or `retryExcludeStatusCodes`.

Recommended Retry Patterns:

- **Status-Code-Based Retry for RESTOutbound:** You must use status-code-based retry for RESTOutbound steps when retries depend on HTTP response status codes. For example

```
"retryConfig": {
 "retryOnlyOnStatusCodes": [502, 503, 504],
 "retryMaxAttempts": 5,
 "retryDelayMilliseconds": 1000,
 "retryBackoffMultiplier": 2.0,
 "retryMaxDelayMilliseconds": 10000,
 "retryJitterMilliseconds": 500,
 "retryTotalTimeoutMilliseconds": 60000
}
```

- **Expression-Based Retry for Business Service or Workflow:** You must use expression-based retry for `BusinessService` or `Workflow` steps when retries depend on a runtime expression. For example:

```
"retryConfig": {
 "retryOnCondition": "${output.'Error Code' == 'TEMPFAIL'}",
 "retryMaxAttempts": 3,
 "retryDelayMilliseconds": 500,
 "retryBackoffMultiplier": 1.0,
 "retryTotalTimeoutMilliseconds": 5000
}
```

response

You can use `response` to define success and error responses for the workflow. `response` must contain both of the following sections:

- `success`
- `error`

Each section supports:

- (Required) `payload`
- (Optional) `statusCode`
- (Optional) `Headers`

The `payload` follows the same structure as step `inputSpecification`.

Error-path behavior:

- For error scenario collection, runtime adds `exception` to the collected payload context when exception information is available.
- Both `statusCode` and `headers` support embedded expressions.

Sample Payload:

```
"response": {
 "success": {
 "payload": {
```

```
"dataCollection": [
{
"source": "Augment_Account_REST",
"transformationSpec": "complex-data-collection.json"
}
],
"transformation": [
{
"transformationSpec": "complex-success-transformation.json"
}
]
},
"statusCode": "${global.rest_status >= 200 && global.rest_status < 300 ? 200 :(global.rest_status >= 400 &&
global.rest_status < 500 ? 400 : 500)}",
"headers": {
"X-Flow-Result": "success",
"X-External-Ref": "${global.external_ref}"
}
},
"error": {
"payload": {
"dataCollection": [
{
"source": "Augment_Account_REST",
"transformationSpec": "complex-data-collection.json"
}
],
"transformation": [
{
"transformationSpec": "complex-error-transformation.json"
}
]
},
"statusCode": 500,
"headers": {
"X-Flow-Result": "error"
}
}
}
```

Transitions

You must use `transitions` to define the workflow state machine. Each key in `transitions` is a state name and contains a `transitions[]` array.

Schema Rules

The following minimum schema rules apply:

- $start is required.
- Each state entry requires `transitions[]`.
- Each transition requires `target`.
- `condition` is optional.
- When `condition` is present, it must use the `${...}` expression format.

Valid targets are:

- Any declared step name
- $end
- $errorEnd

- $start

Runtime Execution Model

Execution begins at $start.

- Runtime evaluates $start.transitions in order.
- The first matching transition decides the next target.
- If the target is a step, runtime executes that step.
- After step execution, runtime evaluates that step state's transitions[] in order.
- Runtime continues until $end or $errorEnd is reached.

The terminal states produce the following responses:

- $end produces `response.success`.
- $errorEnd produces `response.error`

A transition without a condition act as an unconditional fallback.

Condition Evaluation Context

A transition condition can read the following values:

- `WorkflowInput`
- `WorkflowHeaders`
- `Global`
- Step context keys, such as `<StepName>.output`
- On the exception path, `hasExceptionOccurred` and `exception`

Use `global.*` for branch decisions that depend on prior step results. Use `WorkflowInput` and `WorkflowHeaders` for routing based on the original request. Keep one final unconditional transition as a fallback for deterministic routing.

Branching Pattern

The following example shows an ordered if/else branching pattern:

```
"transitions": {
 "$start": {
 "transitions": [
 {
 "target": "Validate_Request"
 }
 ]
 },
 "Validate_Request": {
 "transitions": [
 {
 "condition": "${global.validation_status == 'VALID'}",
 "target": "Fetch_Account"
 },
 {
 "condition": "${global.validation_status == 'INVALID'}",
 "target": "$errorEnd"
 },
 {
 "target": "$errorEnd"
 }
```

```
 ]
 },
 "Fetch_Account": {
 "transitions": [
 {
 "target": "$end"
 }
 ]
 }
}
```

In this pattern, the first true condition wins. The last unconditional transition works as the default `else` branch.

Exception-Aware Routing

You can branch differently on the exception path by using `hasExceptionOccurred` or `exception`.

```
"transitions": {
 "Invoke_External_REST": {
 "transitions": [
 {
 "condition": "${global.hasExceptionOccurred == true}",
 "target": "Build_Error_Response"
 },
 {
 "condition": "${global.rest_status >= 200 && global.rest_status < 300}",
 "target": "$end"
 },
 {
 "target": "$errorEnd"
 }
 ]
 },
 "Build_Error_Response": {
 "transitions": [
 {
 "target": "$errorEnd"
 }
 ]
 }
}
```

Loops and Safety

Loop-back transitions are supported, such as transitions that target an earlier step. Design loop-back transitions with explicit guards. You must use the following approach:

- Maintain loop counters or flags in `global`.
- Add upper-bound checks in conditions.
- Always include a deterministic exit path to $end or $errorEnd.

Pipeline Validation Rules

The pipeline runs validation by using `openint-spec-validator.jar`.

The pipeline validates the following items:

- Base configuration, `config.json`, against `baseConfigSchema.json`.
- Siebel connectivity rules, including `useCG` with the correct URL field.
- UI resource and `loginpath` dependency.

- Domain configuration, `domain-configurations/*.json`, against `integrationConfigSchema.json`.
- Integration path format, including whether the path contains nested paths after `/integrations/<id>`.
- Required `externalDsl`.
- Required `externalREST` and `outboundEventSchema`.
- Merged configuration.
- External DSL schema validation against `externalDSLSchema.json`.
- External DSL semantic validation, including transition targets, expression format, global mapping, and referenced transformation files.
- Step-specific retry rules.
- Additional integration checks.

For the merged configuration check, merged `apigen` must contain at least one of the following:

- `resources`
- `integrations`

Additional Integration Checks

The pipeline enforces the following checks for integration scripts and reusable functions:

- Each `scripts/*.js` file must contain exactly one `executeTransformationScript` function.
- Script files must not contain extra function declarations.
- `.js` file names in `jsonata-configurations/reusable-functions/` must be unique across selected domains.

Failure Artifacts

If validation fails, the pipeline writes failure details to the following files:

```
openint-deploy/failed_config_openapi_keys.txt
openint-deploy/failed_external_dsls.txt
```

Authoring Checklist

Before you package the configuration, verify the following items:

- Add the domain in `config.json` under `apigen.integration-domains`.
- Wire the integration in `domain-configurations/integration.json`.
- Author the DSL according to the schema definition and validation rules.
- Declare every global variable used by `global.<name>` mappings.
- Keep each `transformationSpec` file resolvable under the domain `transformation-spec` folders.
- Ensure that every `restSpec` alias exists in `apigen.externalREST`.
- Ensure that every `queue` alias exists in `apigen.outboundEvents`.
- Optionally validate the configuration before packaging by running one of the following commands:
  `./gen.sh -codegen`

  or
  `gen.bat --codegen`

DSL Schema and Design-Time Validation

Orchestration flows are authored as Domain-Specific Language (DSL) definitions. Each DSL definition must conform to the published DSL schema.

The schema defines the supported DSL structure and semantics, including permitted step types, required attributes, and valid control-flow constructs such as transitions, conditions, loops, and retry definitions. This schema-driven approach helps make integrations consistent and predictable.

Open Integration includes a DSL Validator that runs during the design-time pipeline. The validator checks whether a DSL definition is well formed and schema compliant before it is deployed or executed. It also detects common configuration issues, such as missing required fields, invalid step references, or unsupported constructs.

```
{
 "$schema": "http://json-schema.org/draft-07/schema#",
 "title": "WorkflowConfig",
 "type": "object",
 "required": ["input", "output", "steps", "response", "transitions"],
 "additionalProperties": false,
 "properties": {
 "input": {
 "type": "object"
 },
 "output": {
 "type": "object"
 },
 "global": {
 "type": "array",
 "items": { "type": "string" },
 "description": "Optional list of string global variable names."
 },
 "steps": {
 "type": "object",
 "minProperties": 1,
 "description": "A map (object) of steps. Step names (keys) MUST be unique as per JSON object
 specification.",
 "additionalProperties": {
 "oneOf": [
 {
 "type": "object",
 "required": ["type", "restSpec", "endPoint", "method", "inputSpecification"],
 "description": "RESTOutbound Step.",
 "properties": {
 "type": { "const": "RESTOutbound" },
 "restSpec": { "type": "string" },
 "endPoint": { "type": "string" },
 "method": { "type": "string" },
 "pathParams": { "type": "object" },
 "queryParams": { "type": "object" },
 "inputHeaders": { "type": "object" },
 "inputSpecification": { "$ref": "#/definitions/inputspecification" },
 "retryConfig": { "$ref": "#/definitions/retryConfig" },
 "output": { "$ref": "#/definitions/globalstringmap" },
 "storeInputToGlobal": { "$ref": "#/definitions/globalstringmap" }
 },
 "additionalProperties": false
 },
 {
 "type": "object",
 "required": ["type", "service", "method", "inputSpecification"],
 "description": "Business Step.",
 "properties": {
 "type": { "const": "BusinessService" },
 "service": { "type": "string" },
 "method": { "type": "string" },
```

```
"inputSpecification": { "$ref": "#/definitions/inputspecification" },
"retryConfig": { "$ref": "#/definitions/retryConfig" },
"output": { "$ref": "#/definitions/globalstringmap" },
"storeInputToGlobal": { "$ref": "#/definitions/globalstringmap" }
},
"additionalProperties": false
},
{
"type": "object",
"required": ["type", "serviceName", "inputSpecification"],
"description": "Workflow Step.",
"properties": {
"type": { "const": "Workflow" },
"serviceName": { "type": "string" },
"inputSpecification": { "$ref": "#/definitions/inputspecification" },
"retryConfig": { "$ref": "#/definitions/retryConfig" },
"output": { "$ref": "#/definitions/globalstringmap" },
"storeInputToGlobal": { "$ref": "#/definitions/globalstringmap" }
},
"additionalProperties": false
},
{
"type": "object",
"required": ["type", "queue", "inputSpecification"],
"description": "EventPublish Step",
"properties": {
"type": { "const": "EventPublish" },
"queue": { "type": "string" },
"inputSpecification": { "$ref": "#/definitions/inputspecification" },
"output": { "$ref": "#/definitions/globalstringmap" },
"retryConfig": { "$ref": "#/definitions/retryConfig" },
"storeInputToGlobal": { "$ref": "#/definitions/globalstringmap" },
"inputHeaders": { "type": "object" },
"partitions": {
"type": "array",
"items": { "type": "integer" }
},
"partitionKey": { "type": "string" }
},
"additionalProperties": false
}
]
}
},
"response": {
"type": "object",
"required": ["success", "error"],
"properties": {
"success": { "$ref": "#/definitions/responseCollectorSpec" },
"error": { "$ref": "#/definitions/responseCollectorSpec" }
},
"additionalProperties": false
},
"transitions": {
"type": "object",
"properties": {
"$start": { "$ref": "#/definitions/transitionStep" }
},
"patternProperties": {
"^(?!\\$start).+$": { "$ref": "#/definitions/transitionStep" }
},
"required": ["$start"],
"additionalProperties": false
}
},
"definitions": {
```

```
"inputspecification": {
"type": "object",
"properties": {
"dataCollection": {
"type": "array",
"minItems": 1,
"items": {
"type": "object",
"required": ["source"],
"properties": {
"source": { "type": "string" },
"transformationSpec": { "type": "string" }
},
"additionalProperties": true
}
},
"transformation": {
"type": "array",
"minItems": 1,
"items": {
"type": "object",
"required": ["transformationSpec"],
"properties": {
"transformationSpec": { "type": "string" },
"preTransformationScript": { "type": "string" },
"postTransformationScript": { "type": "string" }
},
"additionalProperties": true
}
},
"transformationScript": { "type": "string" }
},
"required": ["transformation"],
"additionalProperties": true
},
"responseCollectorSpec": {
"type": "object",
"description": "Terminal workflow response collector with payload spec and optional HTTP metadata.",
"properties": {
"payload": { "$ref": "#/definitions/inputspecification" },
"statusCode": {
"description": "Optional HTTP status code for workflow terminal response.",
"oneOf": [
{ "type": "integer", "minimum": 100, "maximum": 599 },
{
"type": "string",
"minLength": 1,
"pattern": "^(\\$\\{[\\s\\S]*\\}|[1-5][0-9]{2})$"
}
]
},
"headers": {
"type": "object",
"description": "Optional response headers map for workflow terminal response.",
"additionalProperties": {
"type": "string",
"minLength": 1
}
}
},
"required": ["payload"],
"additionalProperties": false
},
"globalstringmap": {
"type": "object",
"propertyNames": {
```

```
"type": "string",
"pattern": "^global\\..+"
},
"additionalProperties": {
"type": "string"
},
"description": "A map where each key starts with 'global.' and the value is a string."
},
"transitionStep": {
"type": "object",
"required": ["transitions"],
"properties": {
"transitions": {
"type": "array",
"minItems": 1,
"items": {
"type": "object",
"required": ["target"],
"properties": {
"target": { "type": "string" },
"condition": { "type": "string" }
},
"additionalProperties": false
}
}
},
"additionalProperties": false
},
"retryConfig": {
"type": "object",
"properties": {
"retryMaxAttempts": {
"type": "integer",
"minimum": 1,
"description": "Max number of attempts (including original call)"
},
"retryOnlyOnStatusCodes": {
"type": "array",
"items": { "type": "integer", "minimum": 100, "maximum": 599 },
"description": "Only retry for these HTTP status codes",
"minItems": 1,
"uniqueItems": true
},
"retryExcludeStatusCodes": {
"type": "array",
"items": { "type": "integer", "minimum": 100, "maximum": 599 },
"description": "Do not retry for these HTTP status codes",
"minItems": 1,
"uniqueItems": true
},
"retryOnCondition": {
"type": "string",
"minLength": 1,
"description": "Expression to evaluate if retry should occur"
},
"retryDelayMilliseconds": {
"type": "integer",
"minimum": 0,
"description": "Initial delay (ms) between retries"
},
"retryBackoffMultiplier": {
"type": "number",
"minimum": 1.0,
"default": 1.0,
"description": "Factor for exponential backoff"
},
```

```
"retryMaxDelayMilliseconds": {
"type": "integer",
"minimum": 0,
"description": "Max pause (ms) between retries"
},
"retryJitterMilliseconds": {
"type": "integer",
"minimum": 0,
"description": "Maximum random offset (ms) for retry delay"
},
"retryTotalTimeoutMilliseconds": {
"type": "integer",
"minimum": 1,
"description": "Maximum total time (ms) for all retries"
}
},
"additionalProperties": false,
"allOf": [
{
"anyOf": [
{ "required": ["retryMaxAttempts"] },
{ "required": ["retryTotalTimeoutMilliseconds"] }
]
},
{
"required": ["retryDelayMilliseconds"]
}
]
}
}
}
```

Preparing Domain Configurations

After you define the integration flow, configure the environment-specific settings required for runtime execution.

For example, a DSL might reference an external service, while the actual endpoint URLs, authentication settings, entry points, and event topics are defined separately in the domain configuration.

An Open Integration configuration document, in JSON format, supports the orchestration DSL and defines both runtime and design-time settings in a single structured file. It separates configuration from DSL logic so the same orchestration flow can be reused while environment-specific details, such as endpoints and security, are managed separately.

The Integration Configuration document typically includes the following details:

- Endpoint configuration, including:
  - A list of supported endpoints referenced by the integration.
  - The mapping that links an endpoint to a specific DSL.
- External REST endpoint, or spoke system, details, such as target URLs and connection parameters.
- Authentication and security configuration for external system calls, such as the required authentication type and the corresponding information.
- Eventing configuration, including the publish or subscribe topics used when the integration publishes events.

Use this section to prepare the domain configuration document that connects the integration flow to endpoints, external systems, security settings, and eventing configurations.

By centralizing these details in a JSON document, Open Integration separates orchestration logic from connectivity details. This improves maintainability and simplifies updates when endpoints, credentials, or event topics change.

The base folder for integration configurations is `openint-deploy/integration-configurations`.

Domain Folder Layout

Each domain uses the following folder layout:

```
integration-configurations/
 <Domain>/
 domain-configurations/
 integration.json
 external-dsl/
 *.json
 restoutbound/
 *.json
 jsonata-configurations/
 transformation-spec/
 translated-mappings/
 reusable-functions/
 scripts/
 *.js
```

Domain Configuration File

The `domain-configurations/integration.json` file defines the domain configuration. During the design-time pipeline, `gen.sh` or `gen.bat` merges this file into `merged_config.json`.

The main sections are:

- `apigen.integrations`:

  Use `apigen.integrations` to define integration endpoint paths. Each key is an integration endpoint path in this format `openintegration/v1.0/integration/<name>`.

  Each integration entry includes:
  - Required `externalDsl`
  - Optional `resources.externalREST[]`
  - Optional `resources.inboundEventRegistration`

    Example:

```
{
 "apigen": {
 "integrations": {
 "openintegration/v1.0/integration/checkAccount": {
 "externalDsl": "demo-siebelBS-REST.json"
 },
 "openintegration/v1.0/integration/DBEEventConsume": {
 "externalDsl": "DBE-To-Siebel-Contact-Publish.json",
 "resources": {
 "externalREST": [
 {
 "spokesystem": "describeSystem",
 "describe": "/tmf632/describe",
 "openAPIFilepath": "tm632.json"
 }
 ],
 "inboundEventRegistration": {
 "topic": "OSM-Orders-Produce",
 "partitions": [0],
 "integrationUser": "lapple"
 }
```

```
 }
 }
 }
 }
}
```

Integration Linkage

External DSL files reference these integration configuration sections:

- `apigen.integrations.<path>.externalDsl`
- `apigen.externalREST.<alias>` for `RESTOutbound.restSpec`
- `apigen.outboundEvents.<alias>` for `EventPublish.queue`

- `apigen.externalREST` :

Use `apigen.externalREST` to define external REST aliases used by `RESTOutbound.restSpec` in the DSL.

Each alias requires:

- `baseURL`
- `authentication`

Supported authentication modes are:

- `basic`, with `username` and `password`.
- `oidc-client-credentials`, with `client-id`, `client-secret`, `identity-uri`, `scope`, and `optional oidc-flow`.

- `apigen.outboundEvents` :

Use `apigen.outboundEvents` to define outbound event aliases used by `EventPublish.queue` in the DSL.

Sample:

```
{
 "apigen": {
 "outboundEvents": {
 "dbe-event-publish": {
 "topic": "OSM-Orders"
 },
 "siebel-event-publish": {
 "topic": "SiebelOrderUpdate"
 }
 }
 }
}
```

Open Integration uses this section to publish messages to configured topics.

Create integration.json

You must create `integration.json` from the template. Use the root file `openint-deploy/integrations_template.json` as a starter template when you create a domain configuration.

To create integration.json :

i. Copy `integrations_template.json` to `integration-configurations/<Domain>/domain-configurations/integration.json`
ii. Replace placeholder map keys, such as `#Unique System Name`, with real aliases.

iii. Replace placeholder values with domain-specific values.
iv. Remove template-only guidance fields, such as `#Map` and `#Mandatory`.
v. For each `apigen.externalREST.<alias>.authentication`, configure exactly one authentication mode:
   - `basic`
   - `oidc-client-credentials`

Sample `integration.json`

```
{
 "apigen": {
 "integrations": {
 "openintegration/v1.0/integration/checkAccount": {
 "externalDsl": "demo-siebelBS-REST.json"
 },
 "openintegration/v1.0/integration/DBEEventConsume": {
 "externalDsl": "DBE-To-Siebel-Contact-Publish.json",
 "resources": {
 "externalREST": [
 {
 "spokesystem": "dbe-osm",
 "describe": "/tmf632/describe",
 "openAPIFilepath": "tm632.json"
 }
 ],
 "inboundEventRegistration": {
 "topic": "OSM-Orders-Produce",
 "partitions": [0]
 }
 }
 }
 },
 "externalREST": {
 "castlemock": {
 "baseURL": "https://example.com",
 "authentication": {
 "basic": {
 "username": "user",
 "password": "password"
 }
 }
 },
 "dbe-osm": {
 "baseURL": "https://example.com",
 "authentication": {
 "oidc-client-credentials": {
 "client-id": "client",
 "client-secret": "secret",
 "identity-uri": "https://idcs.example.com",
 "scope": "urn:example:scope"
 }
 }
 }
 },
 "outboundEvents": {
 "dbe-event-publish": {
 "topic": "OSM-Orders"
 },
 "siebel-event-publish": {
 "topic": "SiebelOrderUpdate"
 }
 }
 }
```

```
}
```

Design-Time Schema Validation

At design time, the pipeline validates all domain `domain-configurations/integration.json` files against the Domain Integration Schema JSON.

The schema requires:

- Top-level `apigen`
- `apigen.integrations`
  - Valid integration paths that match `openintegration/<version>/integration/<name>`

Each integration entry requires `externalDsl`.

The optional `resources.externalREST[]` entries require:

- `spokesystem`
- `describe`
- `openAPIFilepath`

The optional `resources.inboundEventRegistration` entry requires:

- `topic`
- `partitions`

Sample:

```
{
 "$schema": "http://json-schema.org/draft-07/schema#",
 "title": "OpenInt Integration Configuration Schema",
 "description": "Schema for validating OpenInt integrations configuration file",
 "type": "object",
 "required": ["apigen"],
 "additionalProperties": false,
 "properties": {
 "apigen": {
 "type": "object",
 "description": "API generation configuration settings",
 "required": ["integrations"],
 "properties": {
 "integrations": {
 "type": "object",
 "description": "Map of integration definitions by their API paths",
 "patternProperties": {
 "^openintegration/(v[0-9]+\\.[0-9]+)/integration/[^/]+$": {
 "type": "object",
 "description": "Individual integration configuration",
 "required": ["externalDsl"],
 "properties": {
 "resources": {
 "type": "object",
 "description": "Resources used by this integration",
 "properties": {
 "externalREST": {
 "type": "array",
 "items": {
 "type": "object",
 "required": ["spokesystem", "describe", "openAPIFilepath"],
```

```
"properties": {
"spokesystem": {
"type": "string"
},
"describe": {
"type": "string"
},
"openAPIFilepath": {
"type": "string"
}
},
"additionalProperties": false
}
},
"inboundEventRegistration": {
"type": "object",
"required": ["topic", "partitions"],
"properties": {
"topic": {
"type": "string"
},
"partitions": {
"type": "array",
"items": {
"type": "integer"
}
}
},
"additionalProperties": false
}
},
"additionalProperties": false
},
"externalDsl": {
"type": "string",
"description": "Name of the external DSL file defining this integration"
}
},
"additionalProperties": false
}
},
"additionalProperties": false
},
"externalREST": {
"type": "object",
"description": "Configuration for external REST API endpoints",
"patternProperties": {
"^.*$": {
"type": "object",
"description": "Configuration for a specific external system",
"required": ["baseURL", "authentication"],
"properties": {
"baseURL": {
"type": "string",
"description": "Base URL for the external system's REST API",
"format": "uri"
},
"authentication": {
"type": "object",
"description": "Authentication configuration for the external system",
"anyOf": [
{
"required": ["basic"]
},
{
"required": ["oidc-client-credentials"]
```

```
}
],
"properties": {
"basic": {
"type": "object",
"description": "Basic authentication credentials",
"required": ["username", "password"],
"properties": {
"username": {
"type": "string",
"description": "Username for basic authentication"
},
"password": {
"type": "string",
"description": "Password for basic authentication"
}
},
"additionalProperties": false
},
"oidc-client-credentials": {
"type": "object",
"description": "OIDC client credentials authentication",
"required": ["client-id", "client-secret", "identity-uri", "scope"],
"properties": {
"client-id": {
"type": "string",
"description": "OAuth2 client ID"
},
"client-secret": {
"type": "string",
"description": "OAuth2 client secret"
},
"identity-uri": {
"type": "string",
"description": "OIDC identity provider URI",
"format": "uri"
},
"oidc-flow": {
"type": "string",
"description": "OIDC authentication flow type",
"enum": ["client-credentials", "user-assertion"]
},
"scope": {
"type": "string",
"description": "OAuth2 scope for the authentication request"
}
},
"additionalProperties": false
}
}
}
},
"additionalProperties": false
}
}
},
"outboundEvents": {
"type": "object",
"description": "Configuration for outbound event publishing",
"patternProperties": {
"^.*$": {
"type": "object",
"description": "Configuration for a specific outbound event topic",
"required": ["topic"],
"properties": {
"topic": {
```

```
"type": "string",
"description": "Event topic name for publishing messages"
}
},
"additionalProperties": false
}
}
}
},
"additionalProperties": false
}
}
}
```

External DSL Files

Use `external-dsl/*.json` files to define integration workflow behavior and step orchestration. For detailed field-level information, see *Preparing DSLs*.

REST Outbound Files

Use `restoutbound/*.json` files to store OpenAPI files for external systems referenced by integrations. These files are referenced by `resources.externalREST[].openAPIFilepath`. Runtime and schema reference flows use these specifications for external payload definitions.

Transformation Specifications

Transformation mappings in Integration Configurations use the JSONata expression language to define how request and response payloads are reshaped.

Transformation spec files are consumed by these External DSL fields:

- `inputSpecification.dataCollection[].transformationSpec`
- `inputSpecification.transformation[].transformationSpec`
- `response.<success|error>.payload.*.transformationSpec`

Store transformation specs in `jsonata-configurations/transformation-spec/*.json`.

Reusable Functions

Use `jsonata-configurations/reusable-functions/*` for shared reusable function files used during transformation execution.

Validation requires `.js` file names to be unique across all selected domains.

Translated Mappings

Use `jsonata-configurations/translated-mappings/*` to provide lookup dictionaries for JSONata transformations.

The JSONata function `getTranslatedMapping(value, type, locale)` uses these mappings to convert source values, such as codes, into translated or display values. If a key is not found, runtime returns the original value unchanged.

Folder contract:

- Location: `integration-configurations/<domain>/jsonata-configurations/translated-mappings/`
- Each file name is treated as a locale, such as `en.json`, `fr.json`, or `ar.json`
- JSON content can contain nested objects
- Runtime flattens nested keys with dot notation

Runtime constructs lookup keys in this format `<type>.<locale>.<value>`.

With the current runtime behavior, `<type>` is the domain folder name under `integration-configurations`.

For example, `integration-configurations/OrderDomain/jsonata-configurations/translated-mappings/en.json`.

Example content:

```
{
 "status": {
 "A": "Approved",
 "R": "Rejected"
 }
}
```

Example JSONata expression:

```
$getTranslatedMapping("status.A",
 "OrderDomain", "en")
```

If the mapping exists, runtime returns `Approved`. If the mapping is missing, runtime returns the original input, such as `status.A`.

Use translated mappings when:

- The output payload must be locale-specific.
- You need shared code-to-label translation logic across multiple transformation specs.
- You want to avoid repeated translation maps in JSONata expressions.

Configuring Dynamic Language for Translated Mappings

You can use Dynamic Language to select a language for each integration request. The resolved language value is available as `global.language` during workflow execution, including JSONata transformations.

Language selection is optional. If you do not configure a language, integrations continue to run as before.

Configuring the Language

The runtime resolves the language from the following sources, in this order:

1. The `language` property in the External DSL file.
2. The `apigen.language` property in the domain `integrations.json` file.

To configure the language, you can:

- Use `apigen.language` to set a default language for the domain:

  ```
  {
   "apigen": {
   "language": "fra"
   }
  }
  ```

- Use the External DSL language property to set a fixed language for an integration:

  ```
  {
   "language": "deu"
  }
  ```

- Use a JEXL expression to derive the language from the initial request context:

  ```
  {
   "language": "${WorkflowHeaders.language}"
  ```

```
}
```

The runtime evaluates the expression before workflow execution. The expression can access the initial request context, including `WorkflowInput`, `WorkflowHeaders`, `username`, `domain`, and initial global values. Step output is not available at this stage.

Language values must match valid mapping file names without the `.json` extension.

Storing Translated Mappings

The translated mappings are stored in a type-specific folder under the domain configuration:

```
integration-configurations/
 OrderDomain/
 jsonata-configurations/
 translated-mappings/
 status/
 enu.json
 fra.json
```

> **Note:** You must use the corresponding Siebel language codes where possible. For example, use `enu.json` for U.S. English and `fra.json` for French. The resolved language value must exactly match the translation file name without the `.json` extension.

For example, the following `status/fra.json` file provides French status labels:

```
{
 "A": "Approved",
 "R": "Rejected"
}
```

Use the mapping type and the resolved request language in the JSONata expression:

```
$getTranslatedMapping("A", "status", global.language)
```

In this example, when `global.language` is `fra`, the runtime reads `translated-mappings/status/fra.json`.

The JSONata expression supplies the mapping type. The runtime does not infer the mapping type.

Runtime Behavior

- If the selected mapping file contains the requested key, the runtime returns the translated value.
- If the mapping file exists but does not contain the requested key, the runtime returns the original input value.
- If the selected mapping file or mapping type folder is missing, the request fails with a controlled translation error.
- If you do not configure a language in the External DSL or as a domain default, the runtime does not create `global.language`.

An explicit language value in the External DSL takes precedence over the domain default. If an External DSL expression returns an invalid, blank, or non-string value, the request fails during initialization. The runtime does not fall back to the domain default.

Validation Summary for Integrations

When `openint-spec-validator` runs, it validates:

- `config.json` against `baseConfigSchema.json`.
  - Domain configuration files against `integrationConfigSchema.json`.
  - The merged config to ensure it contains at least one of `apigen.resources` or `apigen.integrations`.
  - Referenced External DSL files for semantic and schema correctness.
  - Script function contracts in `scripts/`.
  - Reusable function file name uniqueness across selected domains.

Validation failure details are written to:

- `openint-deploy/failed_config_openapi_keys.txt`
- `openint-deploy/failed_external_dsls.txt`

Integration Configuration Checklist

Before you run the pipeline, verify the following items:

1. Add the domain in `config.json` under `apigen.integration-domains`.
2. Create or update `domain-configurations/integration.json`.
3. Ensure every `externalDsl` file exists in `external-dsl/`.
4. Ensure every `RESTOutbound.restSpec` alias exists in `apigen.externalREST`.
5. Ensure every `EventPublish.queue` alias exists in `apigen.outboundEvents`.
6. Ensure every `transformationSpec` file exists under `jsonata-configurations/transformation-spec/`.
7. Keep reusable function file names unique across selected domains.

Configuring config.json

Siebel Open Integration configuration is driven by `config.json`. You can configure the connection details for the Siebel Server and specify the Siebel CRM artifacts that you want to expose through REST, in `config.json`.

The installer creates a template file named `config-template.json`in the `<INSTALL_DIR>/openint-deploy`directory. You must copy the `config-template.json`file in the same folder and rename the copy as `config.json`.

You must configure the following sections in `config.json`:

- `app` : Use the `app`section to configure how Siebel Open Integration connects to the Siebel Server, the port at which Open Integration start, and the path of the transaction metrics.
- `apigen` : Use the `apigen` section to define which Siebel CRM artifacts you want to expose through REST. Siebel Open Integration uses this section to generate and deploy REST APIs based on artifacts in your Siebel application, such as applets, business components, business services, and workflows.
- `eps` : Use `eps` to define EPS and Kafka event settings. It is optional, but when present, it must follow the schema.

The app Section Parameters

You must use the `app` section to configure how Siebel Open Integration connects to the Siebel Server. It includes these parameters:

Parameter	Section	Description
siebel	app	This section includes the information to connect to Siebel Server.

Parameter	Section	Description
useCG	app> siebel	Use this parameter to specify where Siebel Open Integration gets the connection string from. • Set to **true** to fetch the connection string from Cloud Gateway by using **cg-url**. • Set to **false** to use the value in **connect-string**.
cg-url	app> siebel	(Required if the **useCG** parameter is set to **true**.) Use this parameter to specify the Cloud GatewayURL.
connect-string	app> siebel	(Required if the **useCG** parameter is set to **false**.) Use this parameter to specify the connection string that Siebel Open Integration uses to connect to Siebel. This parameter supports both TLS and TCP connection strings. Oracle recommends using TLS-based connection strings to provide secure communication.
loginpath	app> siebel	Use this parameter to specify the relative URLfor the Siebel application that handles UI requests. For example: **/siebel/app/callcenter/enu**.
context	app	Use this parameter to specify the first path segment in Siebel Open Integration REST URLs. For example: **oracle**.
port	app	Use this parameter to specify the port on which the Open Integration server starts. For example: **8433**.
metrics-path	app	Use this parameter to specify the file system location where the Open Integration server stores transactions.
baseURL	app	Use this parameter to specify the base URL used to generate URLs for resources in REST responses.

The apigen Section Parameters

Use the **apigen** section to define which Siebel CRM artifacts you want to expose through REST. You can configure these resources in the **apigen** section:

- The **resources** section to specify the artifacts that Open Integration exposes through REST APIs. Provide each artifact in the appropriate section based on its type:
  - **openintegration/v1.0/ui:** List view and applet pairs to expose UI-based resources. The REST API executes a **ui** query within the specified View and Applet scope. The query follows the user's roles and responsibilities. For example, to fetch data from the View All Account List View and the Applet

SIS Account Entry Applet, specify that View and Applet pair in the ui section as follows: `All Account ListView/SIS Account Entry Applet`.

After you deploy the Open Integration service, you can use the provided URL to fetch data from the server that runs the service. For example: `https://<hostname:port>/<oracle>/openintegration/v1.0/ui/Account/All Account List View/SIS Account Entry Applet`

For more information and examples, see *AppletAPIs (CRUD Operations)*

- `openintegration/v1.0/data`: List business object and business component pairs to access Siebel CRM Business Objects. For example, `Account/Account/`. For more information and examples, see *Business Object APIs* in the *Siebel REST API Guide*.
- `openintegration/v1.0/workflow`: List Siebel workflows that you want to expose as REST APIs. For example, `EAISiebDemo` and `ISS Promotion WS - ApplyProductPromotion - Order`. For more information and examples, see *Workflow APIs* in the *Siebel REST API Guide*.
- `openintegration/v1.0/service`: List Siebel business services that you want to expose through REST APIs. For example, `SiebelAccount/Insert`. For more information and examples, see *Business Service APIs* in the *Siebel REST API Guide*.

- The `apigen.integration-domains` section to specify the Integration Domains associated with the Open Integration project. Add each Integration Domain that the project uses to expose, organize, or manage integration-related artifacts. If the project doesn't use any Integration Domains, you can omit this section.

Note: `apigen.externalREST` and `apigen.outboundEvents` are defined in domain files in `integration-configurations/<Domain>/domain-configurations/`, not in the base `config.json` file.

The eps Section Parameters

You must use the `eps`section to configure the Kafka event settings. It includes the following parameters:

Parameter	Section	Description
SecureAIEgressServer	eps	(Optional) Use this parameter to specify whether the AI Egress server uses secure communication. Set the value to `true` to enable secure communication or `false` to disable it.
SecureAIToKafkaCommunication	eps	Use this parameter to specify whether communication between AI services and Kafka uses secure transport. Set the value to `true` to enable secure communication. When enabled, you must also configure the Kafka keystore, truststore, and password parameters.
AIEgressPort	eps	(Optional) Use this parameter to specify the port on which the AI Egress server listens for connections.
KafkaServers	eps	Use this parameter to specify the Kafka bootstrap servers that Open Integration uses to connect to the Kafka cluster
KafkaKeyStoreName	eps	(Required if SecureAIToKafkaCommunication is set to true.) Use this parameter to specify the keystore file that contains the client certificate and private key used for secure Kafka communication.

Parameter	Section	Description
KafkaKeyStorePassword	eps	(Required if SecureAIToKafkaCommunication is set to true.) Use this parameter to specify the password for the Kafka keystore.
KafkaTrustStoreName	eps	(Required if SecureAIToKafkaCommunication is set to true.) Use this parameter to specify the truststore file that contains the trusted certificates used to validate Kafka server certificates.
KafkaTrustStorePassword	eps	(Required if SecureAIToKafkaCommunication is set to true.) Use this parameter to specify the password for the Kafka truststore.
KafkaPassword	eps	(Required if SecureAIToKafkaCommunication is set to true.) Use this parameter to specify the password used for Kafka authentication.
aieventconfig	eps	This section contains the AI event processing configuration.
ConfigParam	eps > aieventconfig	This section contains the configuration parameters used for AI event processing and Kafka integration.
SynchronousKafkaProduce	eps > aieventconfig > ConfigParam	Use this parameter to specify whether Kafka messages are produced synchronously. Set the value to `true` to wait for message delivery confirmation before continuing processing.
BaseSiebelComponent	eps > aieventconfig > ConfigParam	This section contains the default Siebel component configuration used for event processing.
Alias	eps > aieventconfig > ConfigParam > BaseSiebelComponent	Use this parameter to specify the alias of the Siebel component.
SessionPoolSize	eps > aieventconfig > ConfigParam > BaseSiebelComponent	Use this parameter to specify the number of Siebel sessions maintained in the session pool.
User	eps > aieventconfig > ConfigParam > BaseSiebelComponent	Use this parameter to specify the number of Siebel sessions maintained in the session pool.
Password	eps > aieventconfig > ConfigParam > BaseSiebelComponent	Use this parameter to specify the password for the Siebel user account.
SessionTime	eps > aieventconfig > ConfigParam > BaseSiebelComponent	Use this parameter to specify the session timeout value for Siebel sessions.
NumberOfProducers	eps > aieventconfig > ConfigParam	(Optional) Use this parameter to specify the number of Kafka producer instances created for event publishing.
KafkaProducerMaxRequestSize	eps > aieventconfig > ConfigParam	Use this parameter to specify the maximum request size, in bytes, that a Kafka producer can send.
KafkaTopicConsumerFetchMaxBytes	eps > aieventconfig > ConfigParam	Use this parameter to specify the maximum amount of data, in bytes, that a Kafka consumer can retrieve in a single fetch request.

Parameter	Section	Description
KafkaConsumerPollInterval	eps > aieventconfig > ConfigParam	Use this parameter to specify the interval used by Kafka consumers when polling for messages.
Events	eps > aieventconfig > ConfigParam	This section contains the list of event definitions processed by the AI event framework.
EventStore	eps > aieventconfig > ConfigParam > Events	Use this parameter to specify the event store associated with the event configuration.
NumberOfConsumers	eps > aieventconfig > ConfigParam > Events	Use this parameter to specify the number of Kafka consumer instances assigned to the event.
KafkaConsumerGroupId	eps > aieventconfig > ConfigParam > Events	Use this parameter to specify the Kafka consumer group identifier used by consumers for the event.
SiebelComponent	eps > aieventconfig > ConfigParam > Events	This section contains Siebel component settings that apply specifically to the event configuration.
Alias	eps > aieventconfig > ConfigParam > Events > SiebelComponent	Use this parameter to specify the alias of the Siebel component used for the event.
SessionPoolSize	eps > aieventconfig > ConfigParam > Events > SiebelComponent	Use this parameter to specify the number of Siebel sessions maintained in the session pool for the event.
KafkaConsumerPollInterval	eps > aieventconfig > ConfigParam > Events > SiebelComponent	Use this parameter to specify the polling interval used by the Kafka consumer for the event.
User	eps > aieventconfig > ConfigParam > Events > SiebelComponent	Use this parameter to specify the Siebel user name used to establish sessions for the event.
Password	eps > aieventconfig > ConfigParam > Events > SiebelComponent	Use this parameter to specify the password for the Siebel user account.
SessionTime	eps > aieventconfig > ConfigParam > Events > SiebelComponent	Use this parameter to specify the session timeout value for the event-specific Siebel sessions.
DefaultRetryCount	eps > aieventconfig > ConfigParam	(Optional) Use this parameter to specify the default number of retry attempts after the first processing attempt. This parameter requires **`DefaultRetryBackoff`**. For example, 3.
DefaultRetryBackoff	eps > aieventconfig > ConfigParam	(Optional) Use this parameter to specify the default backoff interval between retries, in seconds. This parameter requires **`DefaultRetryCount`**. For example, 5.
DLQMessagesMaxLimit	eps > aieventconfig > ConfigParam	(Optional) Use this parameter to specify the maximum number of records returned by GET **`/openintegration/v1.0/dlq/messages`**. Default value is 500.

Parameter	Section	Description
EPSMetricsTopic	eps > aieventconfig > ConfigParam	(Optional) Use this parameter to specify the Kafka topic for persistent EPS DLQ metrics. Default value is **eps-metrics**.
KafkaConsumerMaxPollInterval	eps > aieventconfig > ConfigParam > Events[]	(Optional) Use this parameter to specify the Kafka **max.poll.interval.ms** value for this topic, expressed in seconds. For example, 300.
RetryCount	eps > aieventconfig > ConfigParam > Events[]	(Optional) Use this parameter to specify the number of retry attempts after the first processing attempt for this topic. For example, 3.
RetryBackoff	eps > aieventconfig > ConfigParam > Events[]	(Optional) Use this parameter to specify the backoff interval between retries, in seconds, for this topic. For example, 5.
DLQEnabled	eps > aieventconfig > ConfigParam > Events[]	(Optional) Use this parameter to specify whether exhausted eligible failures are written to DLQTopic. Set the value to **true** to enable DLQ. This parameter requires **RetryCount** and **RetryBackoff**.
DLQTopic	eps > aieventconfig > ConfigParam > Events[]	(Required when **DLQEnabled** is set to **true**.) Use this parameter to specify the Kafka topic that receives DLQ records. For example, **eps-dlq**.

Sample config.json

```
{
"app": {
 "siebel": {
 "useCG": false,
 "connect-string": "siebel.tcpip.none.none://<host>:<port>/siebel",
 "loginpath": "/siebel/app/callcenter/enu"
 },
 "context": null,
 "port": 8433,
 "metrics-path": "/path/to/metrics",
 "baseURL": "https://<host>:<port>/"
},
"eps": {
 "SecureAIEgressServer": "false",
 "SecureAIToKafkaCommunication": "false",
 "AIEgressPort": "4002",
 "KafkaServers": "localhost:9092",
 "aieventconfig": {
 "ConfigParam": {
 "NumberOfProducers": "1",
 "SynchronousKafkaProduce": true,
 "Events": [
 {
 "EventStore": "SiebelEvents",
 "NumberOfConsumers": "1",
 "KafkaConsumerGroupId": "consumer"
 }
 ]
 }
 }
},
"apigen": {
```

```
"integration-domains": ["Telco", "Finance"],
"resources": {
"openintegration/v1.0/ui": [
"All Account List View/SIS Account Entry Applet"
],
"openintegration/v1.0/data": [
"Account/Account"
],
"openintegration/v1.0/workflow": [
"EAISiebDemo",
"ISS Promotion WS - ApplyProductPromotion - Order"
],
"openintegration/v1.0/service": [
"Siebel Order/QueryById",
"Siebel Quote/QueryById"
]
"openintegration/v1.0/wsoperation": [
"All"
],
"openintegration/v1.0/workspace": [
"All"
]
}
}
}
```

Note: The `config-template.json` file shows inline help as commented example strings, but JSON doesn't support comments. For the `app` section, replace every placeholder with an actual value. For `apigen`, if you don't want to configure a resource type (for example: workflow), remove the entire entry instead of leaving the placeholder text. Otherwise, the pipeline can fail with an “expected resources are not found” error.

You can use the `ui` section to list View and Applet pairs.

The REST API executes a `ui` query within the specified View and Applet scope. The query follows the user’s roles and responsibilities.

Configuring profile.json

Siebel Open Integration reads its profile configuration from `profile.json`. It defines Siebel Open Integration security settings and the credentials required to communicate with Siebel.

The installer provides a template file named `profile-template.json` in the `<INSTALL_DIR>/openint-deploy` directory.

You must copy `profile-template.json`in the same folder, rename the copy as `profile.json`, and then update it to match your configuration requirements.

See the topic *Managing Security and Authentication* for configuring security related information in `profile.json`.

It includes these parameters:

Parameter	Section	Description
resource-path	Openint > client > tls > private-key > keystore > resource	Use this parameter to specify the name of the JKS file that contains the imported client CA certificate. For example, siebelclientkeystore1.jks

Parameter	Section	Description
type	Openint> client > tls > private-key > keystore	Use this parameter to specify the client private-key keystore type for Open Integration. For example,**JKS**.
passphrase	Openint> client > tls > private-key > keystore	Use this parameter to specify the passphrase required to access the client private-key JKS file. For example, **siebel**.
alias	Openint > client > tls > private-key > keystore > key	Use this parameter to specify the certificate alias that Open Integration uses to authenticate when it makes outbound SSL connections to outbound servers (for example, to Siebel Cloud Gateway).For example: **`siebelclient`**.
alias	Openint > client > tls > private-key > keystore > idp	Specifies the alias of the private key entry used to sign the user-assertion JWT when requesting an OAuth 2.0 token from IDP. Open Integration loads the signing key from the configured keystore by using this alias.
alias	Openint > client > tls > private-key > keystore > idp > oi > key	Specifies the key identifier (**`kid`**) value included in the JWT header during IDP token requests. This value must match the certificate registration configured for the IDP application so that IDP can find the correct public key and verify the JWT signature.
shutdown-grace-period	Openint> server	Use this parameter to specify the time interval (in ISO 8601 duration format) that allows running tasks to finish before the listener shuts down. For example, **PT15S**.
resource-path	Openint> server > tls > trust > keystore > resource	Use this parameter to specify the name of the JKS truststore that contains the imported CA certificates. The server and client use these certificates to validate peer certificates during TLS. For example, **siebeltruststore1.jks.**
type	Openint> server > tls > trust > keystore	Use this parameter to specify the trust keystore type for Open Integration TLS. For example, **JKS**.
passphrase	Openint> server > tls > trust > keystore	Use this parameter to specify the passphrase required to access the trust JKS file. For example, **`siebel`**.
trust-store	Openint> server > tls > trust > keystore	Use this parameter to specify whether the keystore at **resource.resource-path** is used as a truststore. Set to **true** to trust the CA certificates in this JKS file when Open Integration establishes TLS connections (for example, to the Siebel Application Interface).
resource-path	Openint > server > tls > private-key > keystore > resource	Use this parameter to specify the name of the JKS file that contains the imported server CA certificate. For example, **`siebelserverkeystore.jks`**.

Parameter	Section	Description
type	Openint> server > tls > private-key > keystore	Use this parameter to specify the server keystoretype for Open Integration TLS. For example, **`JKS`**.
passphrase	Openint> server > tls > private-key > keystore	Use this parameter to specify the passphrase required to access the keystore JKS file. For example, **`siebel`**.
alias	Openint> server > tls > private-key > keystore > key	Use this parameter to specify the certificate alias that the Open Integration server presents to clients to verify its identity (for example, a browser or Postman). For example: **`siebelserver`**.
path	Openint> features > security > paths	Use this parameter to specify the REST API path. Configure all server-exposed paths that require authentication and authorization. For example, **`/data`**.
methods	Openint> features > security > paths	Use this parameter to specify the supported HTTP methods. For example, **`GET`**, **`POST`**, **`PUT`**, **`DELETE`**, and so on.
authenticate	Openint > features > security > paths > authenticate	Use this parameter to enable or disable OIDC security for this path: • Set to **`true`** to enable OIDC • Set to **`false`** to disable OIDC.
authorize	Openint > features > security > paths > authorize	Use this parameter to enable or disable Attribute-BasedAccess Control (ABAC) security: ·Set to true to enable ABAC. · Set to false to disable ABAC.
scopes	Openint> features > security > paths > abac	Use this parameter to specify the required scope when authorize is set to true. Open Integration authorizes the request only if the OAuthtoken includes a valid scope. Set scopes to the scope value used when you generated the OAuth token. For example: **`read`**.
enabled	Openint> security	Use this parameter to specify the authentication model: • Set to **`true`** to use OAuth. The server uses server.security.features to authenticate and authorize requests. • Set to **`false`** to use perimeter authentication to validate requests.
require-encryption	Openint> security > config	Use this parameter to enable or disable application-wide security encryption: • Set to **`true`** to enable encryption. • Set to **`false`** to disable encryption.
abac	Openint> security > providers	Specifies the ABAC provider. ABAC evaluates attributes (such as the resource and requested action) to determine whether to allow access.

Parameter	Section	Description
identity-uri	Openint> security > providers > oidc	Use this parameter to specify the identity provider URI for the OIDC configuration. For example, the IDCS base URL.
client-id	Openint> security > providers > oidc	Specifies the IDCS OIDC client ID that Open Integration uses to authenticate with the OIDC provider.
client-secret	Openint> security > providers > oidc	Use this parameter to specify the IDCS client secret for the configured OIDC provider. Open Integration uses this secret (along with the clientID) to authenticate with IDCS for OAuth and OIDC flows.
audience	Openint> security > providers > oidc	Use this parameter to specify the IDCS application value for the token consumer. For example: **`openintv2_`**.
server-type	Openint> security > providers > oidc	Use this parameter to specify the OIDC provider type that Open Integration uses. For IDCS, set this value to **`idcs`**.
header-use	Openint> security > providers > oidc	Use this parameter to specify whether Open Integration reads the OAuth access token from the http request header. Set to **`true`** to use the Authorization header.
allow-origins	Openint> restrictive-cors	Use this parameter to specify the list of allowed origins. For example, **`https://foo.com`**, **`https://there.com`**, and so on.
allow-methods	Openint> restrictive-cors	Use this parameter to specify the list of allowed HTTP methods. For example, **`GET`**, **`DELETE`**, and so on.
path-pattern	Openint >cors > paths	Use this parameter to specify the REST path pattern to match. Wildcards are supported. For example, **`"/"`**.
allow-origins	Openint >cors > paths	Use this parameter to specify the list of allowed origins for the matched path. For example, **`**https://foo.com**`**, **`https://there.com`**, and so on.
allow-methods	Openint >cors > paths	Use this parameter to specify the list of allowed HTTP methods for the matched path. For example, **`GET`**, **`DELETE`**, and so on.
enabled	Openint> cache	Use this parameter to enable or disable Coherence caching for user session caching: • Set to **`true`** to connect to the Coherence cache server by using the provided settings. • Set to **`false`** to ignore the cache configuration.
cluster-name	Openint> cache	Use this parameter to specify the Coherence cluster name configured during installation. The default value is **`siebel`**.

Parameter	Section	Description
request-timeout	Openint> cache	Use this parameter to specify the request timeout when connecting to the Coherence server. For example: `2s, 500ms`. The default value is `2s`.
wka	Openint> cache	Use this parameter to specify a comma-separated list of Coherence Well-Known Addresses (WKA). For example, `"127.0.0.1", ** "127.0.0.100\".**`.
tlsEnabled	Openint> cache	Use this parameter to enable or disable TLS for the Coherence client: • Set to `true` to use the Openint.client.tls trust and private key settings. • Set to `false` to ignore the cache TLS settings. .
max-size	Openint > integrations > jsonata > pool	Use this parameter to specify the maximum number of live contexts allowed in the pool. If you don't specify a value, the runtime uses max `(8, CPU cores * 2)`.
core-size	Openint > integrations > jsonata > pool	Use this parameter to specify the number of contexts created at startup and retained as the baseline pool. If you don't specify a value, the runtime uses min `(4, CPU cores)`, capped by max-size.
acquire-timeout-millis	Openint > integrations > jsonata > pool	Use this parameter to specify the maximum time, in milliseconds, to wait when borrowing a context from the pool before the request fails. If you don't specify a value, the runtime uses `250`. The minimum supported value is `50`.
idle-ttl-millis	Openint > integrations > jsonata > pool	Use this parameter to specify the idle lifetime, in milliseconds, after which non-core contexts become candidates for eviction. If you don't specify a value, the runtime uses `300000` (5 minutes). The minimum supported value is `1000`.
cleanup-interval-millis	Openint > integrations > jsonata > pool	Use this parameter to specify the minimum interval, in milliseconds, between idle eviction scans. If you don't specify a value, the runtime uses `30000` (30 seconds). The minimum supported value is `1000`.
ui-connection-threshold	Openint> threshold	Use this parameter to specify the maximum task threshold percentage at which session
enabled	Openint> tracing	Use this parameter to enable or disable tracing: • Set to `true` to enable tracing. • Set to `false` to disable tracing and ignore the tracing configuration.

Parameter	Section	Description
host	Openint> tracing	Use this parameter to specify the host name of the trace collector. For example, **codacollector.coda.svc.cluster.local**.
service	Openint> tracing	Use this parameter to specify the service name shown in the tracing UI for this server. For example, **open-integration**.
otlp-protocol	Openint> tracing	Use this parameter to specify the transport protocol that the Open Telemetry Protocol (OTLP) exporter uses to send telemetry data (traces, metrics, and logs) to the back-end. Valid values are **grpc**, **http/protobuf**, and **http/json**. The default value is **grpc**.
protocol	Openint> tracing	Use this parameter to specify the HTTP protocol to be used when connecting with the tracing server. For example, **http** or **https**.
port	Openint> tracing	Use this parameter to specify the port to use when connecting to the tracing server. For example, **14250**.**.
env	Openint> tracing > tags	Use this parameter to specify the environment tag used to group spans. For example: **development** or **production**.
sampler-type	Openint> tracing	Use this parameter to specify the sampler type the server uses to collect traces. Supported values are **const** and **ratio**.
sampler-param	Openint> tracing	Use this parameter to specify the sampler parameter for the selected sampler type: • If sampler-type is **const**, set to **1** to trace all requests. • Set to any other value to disable tracing. If sampler-type is **ratio**, set a value between **0** and **1**. For example, **0.1** samples 1 in10 requests and **0.001** samples 1 in 1000 requests.
propagation	Openint> tracing	Use this parameter to specify the trace context propagation formats (comma-separated). For example, **w3c**, **b3**, or **jaeger**.
ProfileName	Openint> Profile	Use this parameter to specify a unique profile name used by Siebel Open Integration. For example, **HostFQDN**.
Name	Openint> ConfigParam > Applications	Use this parameter to specify the name of the Siebel application that Siebel Open Integration uses. For example, **callcenter**.
Language	Openint> ConfigParam > Applications	Use this parameter to specify the Siebel application language. For example, **enu**.
ObjectManager	Openint> ConfigParam > Applications	Use this parameter to specify the Siebel Object Manager (OM) for the application. For example, **sccobjmgr_enu**.

Parameter	Section	Description
MaxTasks	Openint> ConfigParam > Applications	Use this parameter to specify the maximum number of active Siebel tasks that Siebel OpenIntegration can use for this application. This value must not exceed the MAX_TASKS component parameter of the associated Siebel Object Manager. For example, `100`.
UserSpec	Openint > Applications > ConfigParam > AuthenticationProperties	Use this parameter to specify the preconfigured user parameter. For example: `SBL_REMOTE_USER`. You can fetch the trust token from SMC. In SMC, go to **Profiles** > **Application Interface** > **Authentication** > **User Specification**.
TrustToken	Openint > Applications > ConfigParam > AuthenticationProperties	Use this parameter to specify the Siebel trust token used to establish the application connection. For example, `ldap` You can fetch the trust token from SMC. In SMC, go to **Profiles** > **Application Interface** > **Authentication** > **Trust Token**
SingleSignOn	Openint > Applications > ConfigParam > AuthenticationProperties	Set this parameter to `true`. Open Integration works only when single sign-on is enabled.
AnonUserName	Openint > Applications > ConfigParam > AuthenticationProperties	Use this parameter to specify the Siebel anonymous user name to use when the actual user's name isn't provided. For example, `GUESTCST`. You can fetch the anonymous user name from SMC. In SMC, go to **Profiles** > **Application Interface** > **Authentication** > **Anonymous User Name**.
SessionTimeout	Openint > Applications > ConfigParam > AuthenticationProperties	Use this parameter to specify the Siebel session timeout (in milliseconds). For example, `900`.
GuestSessionTimeout	Openint > Applications > ConfigParam > AuthenticationProperties	Use this parameter to specify the guest user session timeout (in milliseconds). For example, `300`.
MaxConnections	Openint> ConfigParam > RESTInBound	Use this parameter to specify the maximum number of concurrent connections allowed to Siebel. For example, `20`. The default value is `4`.
TrustToken	Openint> ConfigParam > RESTInBound > RESTAuthenticationProperties	Use this parameter to specify the Siebel trust token used to establish the application connection. For example, `ldap`. You can fetch the trust token from SMC. In SMC, go to **Profiles** > **Application Interface** > **Authentication** > **Trust Token**.

Parameter	Section	Description
AnonUserName	Openint> ConfigParam >RESTInBound > RESTAuthenticationProperties	Use this parameter to specify the Siebel anonymous user name to use when the actual user's name isn't provided. For example, **GUESTCST**. You can fetch the trust token from SMC. In SMC, go to **Profiles** > **Application Interface** > **REST Inbound Authentication** > **Anonymous User Name**.
UserSpec	Openint> ConfigParam > RESTInBound > RESTAuthenticationProperties	Use this parameter to specify the preconfigured user parameter. For example, **SBL_REMOTE_USER**. You can fetch the trust token from SMC. In SMC, go to **Profiles** > **Application Interface** > **REST Inbound Authentication** > **User Specification**.
ObjectManager	Openint> ConfigParam > RESTInBound	Use this parameter to specify the Siebel OM used for inbound REST requests. For example, **eaiobjmgr_enu**.

Sample profile.json

```
{
 "Openint": {
 "client": {
 "tls": {
 "private-key": {
 "keystore": {
 "resource": {
 "resource-path": "siebelclientkeystore.jks"
 },
 "type": "JKS",
 "passphrase": "client",
 "key": {
 "alias": "siebel"
 },
 "idp": {
 "alias": "idp-signing-key",
 "oi": {
 "key": {
 "alias": "openint-key-2026"
 }
 }
 }
 }
 }
 }
 },
 "server": {
 "shutdown-grace-period": "PT15S",
 "tls": {
 "trust": {
 "keystore": {
 "resource": {
 "resource-path": "siebeltruststore.jks"
 },
 "type": "JKS",
 "passphrase": "siebel",
```

```
"trust-store": true
}
},
"private-key": {
"keystore": {
"resource": {
"resource-path": "siebelserverkeystore.jks"
},
"type": "JKS",
"passphrase": "server",
"key": {
"alias": "siebel"
}
}
}
},
"features": {
"security": {
"paths": [
{
"path": "/openintegration/v1.0/[{+path}/]{last:(?!describe$)[^/]+}[/]",
"methods": [
"get",
"post",
"put",
"delete"
],
"authenticate": true,
"authorize": true,
"abac": {
"scopes": [
"read"
]
}
}
]
}
}
},
"cache": {
"enabled": false,
"cluster-name": "siebel",
"request-timeout": "2s",
"tlsEnabled": true,
"wka": [
"127.0.0.1"
]
},
"threshold": {
"ui-connection-threshold": 75
},
"integrations": {
"jsonata": {
"pool": {
"max-size": 16,
"core-size": 4,
"acquire-timeout-millis": 250,
"idle-ttl-millis": 300000,
"cleanup-interval-millis": 30000
}
}
},
"tracing": {
"enabled": false,
"service": "open-integration",
"otlp-protocol": "grpc",
```

```
"protocol": "http",
"host": "localhost",
"port": 4317,
"sampler-type": "const",
"sampler-param": 1,
"propagation": "b3",
"tags": {
"env": "development"
}
},
"security": {
"enabled": false,
"config": {
"require-encryption": false
},
"providers": [
{
"abac": {}
},
{
"oidc": {
"identity-uri": "IDCS URL",
"client-id": "IDCS Client ID",
"client-secret": "IDCS Client Secret",
"audience": "openintv2_",
"server-type": "idcs",
"redirect": false,
"header-use": true
}
}
]
},
"restrictive-cors": {
"allow-origins": [],
"allow-methods": []
},
"cors": {
"paths": [
{
"path-pattern": "",
"allow-origins": [],
"allow-methods": []
}
]
}
},
"Profile": {
"ProfileName": "phoenix21234220441"
},
"ConfigParam": {
"Applications": [
{
"Name": "callcenter",
"Language": "enu",
"ObjectManager": "sccobjmgr_enu",
"MaxTasks": 100,
"AuthenticationProperties": {
"UserSpec": "SBL_REMOTE_USER",
"TrustToken": "ldap",
"SingleSignOn": true,
"AnonUserName": "GUESTCST",
"SessionTimeout": 300,
"GuestSessionTimeout": 300
}
}
],
```

```
"RESTInBound": {
"MaxConnections": 20,
"RESTAuthenticationProperties": {
"TrustToken": "ldap",
"AnonUserName": "GUESTCST",
"UserSpec": "SBL_REMOTE_USER"
},
"ObjectManager": "eaiobjmgr_enu"
}
}
}
```

Note: The `profile-template.json`file includes inline help text as commented lines inside double quotes to guide you when you update `profile.json`. Because JSON doesn't support comments, each key must have a valid value. You must replace the help text in double quotes with the actual values for your environment.

Note: When you configure secured route matching under `Openint.server.features.security.paths`, specify route templates rather than runtime values. For example, use `/openintegration/v1.0/service/Account/{Id}` instead of `/openintegration/v1.0/service/Account/12345`.

Configuring Maven Settings

Siebel Open Integration is a Helidon SE–based Maven project. Siebel Open Integration generates code on the fly and then uses Maven to build, package, and create the JAR artifacts.

The Siebel Open Integration installer generates a Maven settings template file named `settings.xml.template` in `<INSTALL_DIR>`.

To configure the Maven local repository, perform the following steps:

1. Copy the `settings.xml.template` file in the same folder.
2. Rename the copied file as `settings.xml`.
3. Edit the `<localRepository>` parameter in the `settings.xml` file to point to the bundled compile dependency folder:

   ```
   <localRepository><installed_location>\compile_dependency\repository</localRepository>
   ```

   In the example above, `<installed_location>` is the Siebel Open Integration installation directory.

Running the Design Time Pipeline

After you configure `config.json`and `profile.json`, run the design-time pipeline by using `gen.sh`. Ensure that the Siebel application is running when you run the design-time pipeline.

The design-time pipeline performs these tasks:

- Downloads configurations from Git.
- Reads configuration from `config.json` and `profile.json`.
- Connectsto Siebel and retrieves metadata for the resources configured in `config.json`.
- Generates the OpenAPI specification from the retrieved metadata.
- Generates Plain Old Java Objects (POJOs) that bind to the metadata.
- Generates the service implementation code that exposes the REST endpoints.

- Builds the Helidon service so it is ready to start and process requests.

To run the design-time pipeline script:

1. Navigate to the `openint-deploy` directory:
   ```
   cd <INSTALL_DIR>/openint-deploy
   ```
2. Run the design-time pipeline script:
   ```
   gen.sh
   ```
   On Windows, run:
   ```
   gen.bat
   ```

At the end of execution, the pipeline prints a pipeline error summary:

- Creates an `error-summary.log` file in the current folder to capture the execution summary.
- Creates an `error.log` file in the current folder with detailed information if errors occur.

You can use the following options with the design-time pipeline script to control code generation, packaging, startup, OpenAPI logging, and Git-based configuration download:

Option	Description
--codegen	Runs code generation only. Generates and validates artifacts without packaging. This option is useful for the IDE experience.
--package	Runs packaging only. Packages from existing src/main/resources/ outputs.This option expects outputs from a prior --codegen run.
--start	Starts the server (target/openint.jar) after packaging and redirects runtime output to server_ output.log.
--openApiSpecLogLevel:<LEVEL>	Sets the OpenAPI retriever log level. Supported values are TRACE, ALL, ERROR, INFO, FATAL, DEBUG, OFF, and WARN.
--download-from-git	Pulls configurations by using git.json before configuration validation.
--git-token=<token>	Exports GIT_TOKEN for Git HTTP or HTTPS authentication.
--git-trsusttore-password=<password>	Exports GIT_TRUSTSTORE_PASSWORD for HTTPS truststore access. This option is strongly recommended for .jks, .p12, and .pfx files.
--clean-configurations	With --download-from-git, replaces local integration-configurations from the Git source.

For example, to download configurations from Git before packaging:

- On Windows:
  ```
  .\gen.bat
   -download-from-git
  ```
- On Linux:
  ```
  ./gen.sh
   --download-from-git
  ```

Note: Run the design-time pipeline each time you update `config.json` and `profile.json`. This ensures the service is rebuilt based on the latest specification.

For details about the script and its supported parameters, see *Design Time Workflow and Supporting Artifacts*.

End-to-End Design-Time Pipeline Flow

The pipeline validates configuration inputs, initializes the runtime environment, generates code, packages the generated content, and optionally starts `target/openint.jar`. The pipeline is staged, and multiple stages as follows:

1. **Validation and Initialization Stage**

   During validation and initialization, the pipeline performs the following tasks:

   - Initializes logs:
     - `error.log`
     - `error-summary.log`
     - `server_output.log`
   - Performs mandatory prerequisite checks for the following files:
     - `jq` or `jq.exe`
     - `parent-pom.xml`
     - `settings.xml`
     - `openint-dsl.jar`
     - `openapi-generator-cli.jar`
   - Performs mandatory prerequisite checks for the following folders:
     - `graalvm-jdk`
     - `apache-maven`
     - `compile_dependency`
     - `openapi-generator`
   - Sets up the environment:
     - `JAVA_HOME`
     - `MAVEN_HOME`
     - Local Maven repository
   - Checks for configuration location ambiguity between the root location and `src/main/resources`.
   - Validates JSON syntax for configuration files.
   - Performs Open Integration specification validation by using `openint-spec-validator.jar`.

   The scripts use the following fail-fast markers:

   - BASE_CONFIG_VALIDATION_FAILED
   - APIGEN_VALIDATION_FAILED

2. **Code Generation Stage**

   The code generation stage performs the following tasks:

   - Removes previously generated content.
   - Runs Maven install for the required local JAR files.
   - Retrieves OpenAPI specifications by using `openapi-spec-retriever.jar`.
   - Merges `describe/*.json` into `merged.json` as follows: `openint-dsl.jar mergedescribe`
   - Generates the server scaffold by using `openapi-generator-cli.jar`.
   - Generates the external DSL model as follows: `openint-dsl.jar externaldsl`
   - Generates the internal DSL as follows: `openint-dsl.jar internaldsl`
   - Generates the integration workflow as follows:`openint-dsl.jar integrations …`

   If the `describe/` directory is empty but `event_publish_baseline_openapi.json` exists, codegen uses `event_publish_baseline_openapi.json` as the baseline file.

   When the stage is run with the `–codegen` option, the pipeline moves the following runtime configuration assets to `src/main/resources/`:

   - `log4j2.xml`
   - `merged_config.json`
   - `profile.json`
   - `integration-configurations/`
   - `*.jks`

3. **Package Stage**

   The package stage uses configurations from `src/main/resources/` and performs the following tasks:

   - Runs the Maven package build.
   - Copies resources to target/.
   - Copies integration configurations to `target/`.
   - Cleans up transient build directories in `target/`.

   The package stage creates the `target/openint.jar` output artifact.

Optional Start

If `--start` is set, the pipeline starts the JAR file by running: `java $JAVA_OPTS -jar target/openint.jar`.

Before you start the JAR file, ensure that all configured spoke systems, including Kafka, are up and running.

Configuration Location Rules

You must keep all configuration inputs in one base location for each run. Use one of the following locations:

- `openint-deploy/`
- `openint-deploy/src/main/resources/`

Note: You must not split files or folders across both locations in the same run. Splitting configuration inputs across both locations causes pipeline failure. If this occurs, clean up the configurations and rerun the pipeline.

Required Inputs

The pipeline requires the following core files and folders:

- `config.json`
- `profile.json`
- `log4j2.xml`
- `integration-configurations/`
- `JKS files`

Artifacts and Logs

The pipeline creates or uses the following artifacts:

- `target/openint.jar`
- `describe/*.json`
- `merged.json`
- `merged_config.json`

The pipeline writes the following logs:

- `error.log`
- `error-summary.log`
- `server_output.log`

Deploying Siebel Open Integration

This topic describes how to deploy Siebel Open Integration.

After the design-time pipeline completes successfully, you can deploy the Siebel Open Integration service by running the start script as follows:

- On Linux:
  `<INSTALL_DIR>/start.sh`
- On Windows:
  `<INSTALL_DIR>/start.bat`

When the service starts, the console displays a confirmation message indicating that the Open Integration service is running and ready to accept REST requests. You can now use any REST API client to send requests.

Kubernetes Deployment

To deploy Siebel Open Integration on a Kubernetes Cluster using SCM, refer to the *Deploying Open Integration on a Kubernetes Cluster Using SCM* in *Deploying Open Integration on a Kubernetes Cluster Using Siebel Cloud Manager Guide*.

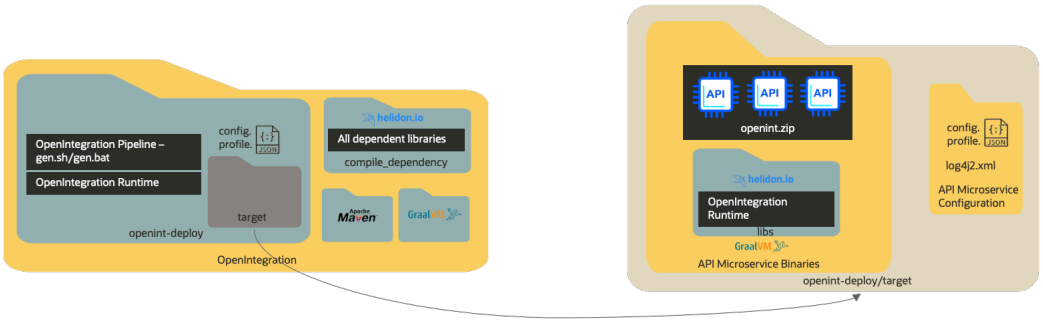
Design Time Workflow and Supporting Artifacts

Siebel Open Integration projects include the following primary artifacts:

- `openint_deploy`: Use `openint_deploy` as the main project folder. It contains configuration files, generated source, compilation output, and dependencies. It includes the following files and directories:
- `config-template.json`: Template file that you can copy as `config.json` to specify Siebel Server connection details and the Siebel CRM artifacts to expose through REST APIs.

  For more information about `config.json`, see *Configuring config.json* .
- `profile-template.json`: Template file that you can copy as `profile.json` to define Siebel Open Integration security settings and the credentials required to communicate with Siebel CRM.

  For more information about `profile.json`, see *Configuring profile.json*.
- `gen.bat` or `gen.sh`: Run this script to execute the design-time pipeline and build the executable application.
  - `target`: Created after the pipeline completes. This directory contains the following files and directories:
    - `libs`: Contains third-party runtime dependencies.
    - `logs`: Contains log files generated during execution.
    - `config.json`: Copied from the configured source.
    - `profile.json`: Copied from the configured source.
    - `log4j2.xml`: Controls logging levels and severity.
    - `openint.jar`: The generated executable file.
- `start.bat` or `start.sh`: Use this script to start the Open Integration application.
- `compile_dependency`: Contains compile-time and runtime dependencies. These dependencies are available locally and do not require additional downloads.
- `apache-maven`: The build tool used for the Open Integration application.
- `settings.xml.template`: A template file that you must rename to `settings.xml`.
- `graalvm-jdk`: The GraalVM Java Development Kit (JDK) bundled for compilation and runtime execution.



Managing Security and Authentication

Siebel Open Integration helps you expose Siebel CRM resources as REST APIs while aligning with common security standards. Managing security and authentication in Siebel Open Integration involves configuring design-time artifacts and runtime components that control how the application is built, secured, and executed.

To secure the deployment, you must configure the following:

- Siebel Open Integration security components.
- Siebel Business Applications. For more information on securing Siebel Business Applications, refer to the *Siebel Security Guide*.
- The underlying computing environment.

The `profile.json` file is the central configuration file for application and security settings. It is created based on `profile-template.json`. For more information, see *Configuring Siebel Open Integration*.

Configuring Security for Siebel Open Integration

To configure security for Siebel Open Integration, you must:

- Configure server security.
- Configure user authentication for secure access.
- Configure API authorization for API access control.
- Configure network encryption for data confidentiality.
- Encrypting sensitive information such as secrets, in configuration files.
- Configure support for Cross-Origin Resource Sharing (CORS).

Configuring Server Security

You can configure server security in the `profile.json` file using the `client` and `server` sections:

- The `server` section defines TLS settings for inbound connections. It specifies the truststore and server keystore used to validate certificates and present the server identity.
- The `client` section defines TLS settings for outbound connections. It defines the client keystore used by Siebel Open Integration when it establishes secure outbound (SSL/TLS) connections to external systems. The `client.idcs` section contains the Oracle Identity Cloud Service (IDCS) settings required to obtain OAuth 2.0 JSON Web Token (JWT) access tokens.

Note: You must ensure that the keystore and truststore JKS files are available in the configured path.

Note: If you are using a single certificate for both server and client authentication, you can continue to use it until it expires. In this case, configure both the `client` and `server` sections in `profile.json` to reference the same keystore that contains this certificate. After you renew the certificate, update the `client` and `server` sections with the renewed certificate information.

For more information about the sections and parameters in the `profile.json` file, see *Configuring profile.json*.

Configuring User Authentication

Siebel Open Integration requires SSO based authentication for secure access. It only works with Siebel instances that are configured for SSO. You must ensure that the following components are configured for SSO:

1. Object Managers that host resources exposed as APIs.
2. Siebel REST APIs.
3. User interface (UI) applications in the Application Interface profile.
4. Siebel Gateway in the Gateway Security profile.

To enable SSO for a Siebel instance:

1. Configure all Object Managers that host resources exposed as APIs to use SSO.
2. Configure the Siebel Gateway to use SSO in the Gateway Security profile.

Siebel Open Integration supports the following user authentication methods:

Enabling OAuth Authentication

OAuth authentication in Siebel Open Integration uses JSON Web Token (JWT). This is the default authentication method.

You can generate an OAuth 2.0 based JWT token and include it in REST requests. The token carries authentication and authorization information required to access Siebel resources. Siebel Open Integration uses an OpenID Connect (OIDC) provider to validate JWT tokens. By default, it uses the Helidon SE OIDC provider. For details about OIDC provider properties, see the *Implemented Security Providers* section in the Helidon online documentation.

The `profile.json` file defines the security configuration for the Siebel Open Integration server. You must update this file to match your environment by configuring the required authentication mechanisms and security settings, as described in the *Configuring profile.json* section.

To configure OAuth authentication for Siebel Open Integration:

1. Enable SSO for Siebel instance as discussed above.
2. Enable OAuth or JWT authentication through OIDC user assertion token or client credentials in `profile.json`, as follows:
   a. In the `Openint > server > security` section, set the `enabled parameter` to `true`.
   b. In the `Openint > server > features > security` section, configure REST paths:
      i. Set the `path` parameter to REST path. For example, `/*`, `/data/*`, `/service/*`.
      ii. Specify `methods` such as `get`, `post`, `put`, and `delete`.
      iii. Set the `authenticate` and `authorize` parameters:
         - Set `authenticate` to `true` and `authorize` to `true` for user assertion token flow.
         - Set `authenticate` to `true` and `authorize` to `false` for client credentials flow.
   c. In the `Openint > server > security > providers -> oidc section`, configure the following parameters as per your deployment:
      - `identity-uri`
      - `client-id`
      - `client-secret`
      - `audience`
   d. Keep all other parameters unchanged unless your deployment requires updates.

Note: Use the sample configuration to apply OIDC security to all Siebel Open Integration REST paths. This ensures that OAuth authentication is enforced consistently across all endpoints.

Note: In a client credentials grant flow, the client ID is used as the subject. You must create a Siebel user or employee with the client ID as the `USERID` and assign the required access permissions to that user. For more information, refer the *Siebel Security Guide*.

For a sample `profile.json` configuration, see *Configuring profile.json*.

Configuring Perimeter Authentication

You can use perimeter authentication in production only when an API Gateway performs authentication and forwards the authenticated user identity in an HTTP header. Siebel Open Integration reads the header specified in `UserSpec` and uses it as the current subject's principal.

To configure perimeter authentication in Siebel Open Integration:

1. Enable SSO for the Siebel instance.
2. Set `UserSpec` to `SBL_REMOTE_USER`. You must configure `UserSpec` for both UI (Applications) and non-UI REST (RESTInbound) sections.
3. Set the `authenticate` parameter to `false` for any Open Integration REST path that should use perimeter authentication.
4. Use perimeter authentication (set `authenticate` to `false`) for `/ui` path while keeping OIDC enabled (set `authenticate` to `true`) for the other REST paths.

```
{
 "path": "/ui*",
 "methods": [
 "get",
 "post",
 "put",
 "delete"
 ],
 "authenticate": false
 }
```

For a sample `profile.json` configuration, see *Configuring profile.json*.

Configuring API Authorization

Siebel Open Integration provides access control for REST APIs using the ABAC authorization provider from Helidon SE. The ABAC provider works together with the OIDC provider to enforce authorization based on token scopes.

For more information about ABAC provider configuration, refer *ABAC Provider* section in the online Helidon documentation.

The ABAC provider must be used together with the OIDC provider to enable API authorization. You must add the ABAC provider under the `security.providers` section in `profile.json` and define scope validation rules for each REST path under the `server.features.security.paths` section.

To configure the OIDC security with OAuth under each REST path:

1. Set `authorize` to `true` to enable ABAC authorization.

   Note: To configure perimeter access, if required, set `authenticate to false` and `authorize to false` for paths that use perimeter authentication.

2. Specify HTTP methods (`get`, `post`, `put`, and `delete`). Only the listed methods are secured. Methods not listed are not protected.
3. Set `"scopes"` to match the values used during OAuth token generation.
4. Add the `abac` section as follows:

```
"paths": [
 {
 "path": "/data*",
 "methods": [
 "get",
 "post",
 "put",
 "delete"
 ],
 "authenticate": true,
 "authorize": true,
 "abac": {
 "scopes": [
 "read"
 ]
 }
 }
]
```

Sample configuration for the ABAC security for Siebel Open Integration REST paths.

```
{
 "Openint": {
 "server": {
 "shutdown-grace-period": "PT15S",
 "tls": {
 "trust.keystore": {
 "resource.resource-path": "siebeltruststore1.jks",
 "type": "JKS",
 "passphrase": "siebel",
 "trust-store": true
 },
 "private-key.keystore": {
 "resource.resource-path": "siebelkeystore1.jks",
 "type": "JKS",
 "passphrase": "siebel",
 "key.alias": "siebel"
 }
 },
 "features": {
 "security": {
 "paths": [
 {
 "path": "/data*",
 "methods": [
 "get",
 "post",
 "put",
 "delete"
 ],
 "authenticate": true,
 "authorize": true,
```

```
"abac": {
"scopes": [
"read"
]
}
},
{
"path": "/service*",
"methods": [
"get",
"post",
"put",
"delete"
],
"authenticate": true,
"authorize": true,
"abac": {
"scopes": [
"read"
]
}
},
{
"path": "/workflow*",
"methods": [
"get",
"post",
"put",
"delete"
],
"authenticate": true,
"authorize": true,
"abac": {
"scopes": [
"read"
]
}
},
{
"path": "/ui*",
"methods": [
"get",
"post",
"put",
"delete"
],
"authenticate": false,
"authorize": false,
"abac": {
"scopes": [
"read"
]
}
}
]
}
}
},
"security": {
"enabled": true,
"config.require-encryption": false,
"providers": [
{
"abac": {
"fail-on-unvalidated": true,
"fail-if-none-validated": true
```

```
}
},
{
"oidc": {
"identity-uri": "IDCS URL",
"client-id": "IDCS Client ID",
"client-secret": "IDCS Client Secret",
"audience": "openintv2_",
"server-type": "idcs",
"redirect": false,
"header-use": true
}
}
]
}
}
}
```

Configuring Network Encryption

Siebel Open Integration supports HTTPS to ensure data confidentiality during communication with external systems.

To configure Siebel Open Integration to accept and send HTTPS request:

1. Import the Siebel Application Interface CA certificate into the truststore:
   a. Add the certificate to `siebeltruststore.jks`.
   b. Reference this file in `server.keystore.resource.resource-path`.
2. Generate a server certificate for Siebel Open Integration:
   a. Import the certificate into `siebelkeystore.jks`.
   b. Reference this file in `private-key.keystore.resource.resource-path`.
3. Create an SSL certificate for the Siebel Application Interface. For the steps to create an SSL certificate, refer the *About Certificates and Key Files Used for TLS Authentication* section in *Siebel Security Guide*.
4. Copy the keystore and truststore JKS files in the `openint-deploy` directory.
5. Update the `profile.json` file to configure the following under the `server` section:
   a. Configure HTTPS port.
   b. Configure truststore and keystore paths.
   c. Provide keystore passphrase and key alias.

Encrypting Configuration Files

Siebel Open Integration supports encryption of sensitive values such as passwords in the `profile.json` file.

To enable encryption:

1. Set environment variable `SECURE_CONFIG_AES_MASTER_PWD` to define the master encryption password. For example:
   ```
   export SECURE_CONFIG_AES_MASTER_PWD=openint
   ```
2. Use the Helidon encryption utility to encrypt sensitive values:
   ```
   java -cp "target/libs/*" io.helidon.config.encryption.Main aes masterPassword secretToEncrypt
   ```
   In the above example:
   - `<masterPassword>` is the value of `SECURE_CONFIG_AES_MASTER_PWD`.
   - `<secretToEncrypt>` is the value to encrypt.

For example:

```
java -cp "target/libs/*" io.helidon.config.encryption.Main aes openint siebel
```

Generated encrypted string is

```
${GCM=aUP6KWK2mC4kfKrVMPfV5RVPRk5l65rGWy7ocyyDyYl55WnwgW4vozlVopsZyvTWL/Q=}
```

Generate the encrypted client-secret value for the client-secret "441a09a3-21bd-4d26-bcc7-ce681cca93f6"

```
java -cp "target/libs/*" io.helidon.config.encryption.Main aes openint 441a09a3-21bd-4d26-bcc7-
ce681cca93f6
```

Generated encrypted string is

```
${GCM=3p9/VJX6Emls9MWTE8a7u41vf6Nb/ VwGD7SYnoytykWwCv81t4Sg1zQiOg6c2mfBw3E9if27Ju8XW4h/
PN8fyacTUccDHIzmHwCksCrpwyE=}
```

3. Update `profile.json` as follows:
   a. Set `config.require-encryption` to `true` under the `security` section.
   b. Replace plain text values in `profile.json` with encrypted values. For example:

```
{
 "Openint": {
 "server": {
 "shutdown-grace-period": "PT15S",
 "tls": {
 "trust.keystore": {
 "resource.resource-path": "siebeltruststore1.jks",
 "type": "JKS",
 "passphrase": "${GCM=aUP6KWK2mC4kfKrVMPfV5RVPRk5l65rGWy7ocyyDyYl55WnwgW4vozlVopsZyvTWL/Q=}",
 "trust-store": true
 },
 "private-key.keystore": {
 "resource.resource-path": "siebelkeystore1.jks",
 "type": "JKS",
 "passphrase": "${GCM=aUP6KWK2mC4kfKrVMPfV5RVPRk5l65rGWy7ocyyDyYl55WnwgW4vozlVopsZyvTWL/Q=}",
 "key.alias": "${GCM=aUP6KWK2mC4kfKrVMPfV5RVPRk5l65rGWy7ocyyDyYl55WnwgW4vozlVopsZyvTWL/Q=}"
 }
 }
 },
 "security": {
 "enabled": true,
 "config.require-encryption": true,
 "providers": [
 {
 "oidc": {
 "identity-uri": "********",
 "client-id": "********",
 "client-secret": "${GCM=3p9/VJX6Emls9MWTE8a7u41vf6Nb/
VwGD7SYnoytykWwCv81t4Sg1zQiOg6c2mfBw3E9if27Ju8XW4h/PN8fyacTUccDHIzmHwCksCrpwyE=}",
 "audience": "********",
 "server-type": "idcs",
 "redirect": false,
 "header-use": true
 }
 }
 ]
 }
 }
}
```

4. Restart Siebel Open Integration for changes to take effect.

For more information on encryption, refer the *Encrypting values* section in the *Helidon online documentation*.

Configuring Support for CORS

Siebel Open Integration supports Cross-Origin Resource Sharing (CORS) to control how REST resources are accessed from different origins.

By default, CORS is enabled for all origins and HTTP methods. To restrict access, you can define CORS rules in the `profile.json` configuration file. Add the CORS configurations after the `security` section.

Before configuring CORS, determine the required access level for each resource. For example, you might allow unrestricted access for simple requests (GET, POST) and restrict other methods (PUT, DELETE) to specific origins (for example, test1.com and test2.com).

Your application can implement three CORS behaviors:

- Simple
  - No configuration required. A relaxed policy for simple requests.
  - Allows all origins and HTTP methods.
- Restrictive (`restrictive-cors`)
  - Use the `restrictive-cors` section to define global restrictions (a stricter policy for certain requests).
  - Sample `restrictive-cors` configuration in `profile.json`:

```
"restrictive-cors": {
 "allow-origins": [
 "https://foo.com",
 "https://there.com"
 ],
 "allow-methods": [
 "GET",
 "DELETE"
 ]
}
```

    This configuration allows only the specified origins to access the listed HTTP methods.
- Overrides (`cors`)
  - This configuration allows only the specified origins to access the listed HTTP methods.
  - Sample `cors` configuration in `profile.json`:

```
{
 "restrictive-cors": {
 "allow-origins": [
 "https://foo.com",
 "https://there.com"
 ],
 "allow-methods": [
 "GET",
 "DELETE"
 ]
 }
 "cors": {
 "paths": [
 {
 "path-pattern": "/",
 "allow-origins": [
 "https://foo.com",
```

```
"https://there.com",
"https://test3.com"
],
"allow-methods": [
"GET",
"DELETE"
]
}
]
}
}
```

This override allows access for `test3.com` in addition to `test1.com` and `test2.com`.

CORS rules are evaluated in the following order:

1. Simple (default behavior)
2. Restrictive (`restrictive-cors`)
3. Overrides (`cors`)

Configuring Open Integration Enterprise Cache for Siebel

This topic describes how you can configure Enterprise Cache for Open Integration in SMC. You can use a non-Advanced configuration when SMC manages the required cache configuration, or an Advanced configuration when you need to add or update the required Open Integration cache XML.

> **Note:** This topic covers Open Integration cache configuration only. For Enterprise Cache installation, cluster topology, hosts, ports, sizing, and general Siebel Management Console (SMC) deployment procedures, refer the *Introduction to Coherence Cache* documentation.

It includes the following sections:

- *Prerequisites for Advanced Configuration*
- *Before You Begin*
- *Configuring Advanced Cache*
- *Cache Configuration Scenarios*
- *Validating the Deployment*
- *Recovering from Coherence Request Timeout Errors*
- *Configuring the Open Integration Cache Request Timeout*
- *Advanced XML Reference*

Prerequisites for Advanced Configuration

When you use an Advanced configuration, you must ensure that all three Open Integration cache mappings and all three corresponding schemes are present in the XML configuration. Do not configure only one mapping or scheme.

Purpose	Cache Name Pattern	Scheme	Service or Listener
Open Integration sessions	`OpenInt_*`	`oi-distributed-sessions`	`DistributedCache / OpenIntEvictionListener`
Open Integration REST sessions	`OpenIntRest_*`	`oi-distributed-rest-sessions`	`DistributedCache / OpenIntRestEvictionListener`
Open Integration current-task coordination	`OpenIntAtomic_*_CurrentTasks`	`oi-concurrent`	`DistributedConcurrentCache`

The configuration must also include `oIExtendTcpProxyService` for Open Integration.

Note: Names are case-sensitive. Use `OpenIntRest_*` exactly as shown. Do not use `OpenIntRest\_*`.

Before You Begin

For an existing environment, you must complete the following steps before you change or redeploy Enterprise Cache:

1. Stop Open Integration.
2. Complete the applicable cache configuration and deployment steps.
3. Confirm that the Enterprise Cache Server deployment completes successfully.
4. Start Open Integration.

Note: Redeploying Enterprise Cache clears existing cache and session entries. This behavior is expected during an upgrade because Siebel restart also invalidates these entries.

Configuring Advanced Cache

To use an Advanced configuration:

1. In SMC, open **Profiles**, and then select **Enterprise Cache Server**.
2. Under **Cache Configuration**, enable **Advanced**.
3. Add or update the required Open Integration cache XML.

You don't need to make Open Integration-specific changes under **Tango Coherence Override**.

Cache Configuration Scenarios

This section describes the different scenarios for advanced and non-advanced cache configurations. It includes the following sections:

- *Scenario 1: New Installation with Non-Advanced Cache Configuration*
- *Scenario 2: New Installation with Advanced Cache Configuration*
- *Scenario 3: Upgrade with an Existing Advanced Cache Configuration*
- *Scenario 4: Upgrade with an Existing Non-Advanced Cache Configuration*
- *Scenario 5: Upgrade Without Existing Cache Using a Non-Advanced Configuration*
- *Scenario 6: Upgrade Without Existing Cache Using an Advanced Configuration*

Scenario 1: New Installation with Non-Advanced Cache Configuration

You can use this scenario for a new Siebel CRM environment that uses a non-Advanced cache configuration.

To perform a new installation using a non-Advanced cache configuration:

1. Create a new non-Advanced Enterprise Cache server profile in SMC.
2. Configure the Enterprise Cache server profile. For more information, refer the *Introduction to Coherence Cache* documentation.
3. Deploy the Enterprise Cache server profile.
4. Confirm that the deployment completes successfully.
5. Start Open Integration.

Note: No custom Open Integration cache XML is required.

Scenario 2: New Installation with Advanced Cache Configuration

You can use this scenario for a new Siebel CRM environment that requires an Advanced cache configuration.

To perform a new installation using an Advanced cache configuration:

1. Create an Enterprise Cache Server profile in SMC.
2. Under **Cache Configuration**, enable **Advanced**.
3. Configure Enterprise Cache installation and general settings. For more information, refer the *Introduction to Coherence Cache* documentation.
4. Add the required Open Integration Advanced configuration, all three cache mappings, all three schemes, both eviction listeners, `oIExtendTcpProxyService`, and `ExtendTcpProxyService`.
5. Deploy the Enterprise Cache Server profile.
6. Confirm that the deployment completes successfully.
7. Start Open Integration.

Scenario 3: Upgrade with an Existing Advanced Cache Configuration

You can use this scenario when the earlier release already uses an Advanced Enterprise Cache Server profile.

Siebel 26.8 adds a dedicated cache for Open Integration REST sessions.

To upgrade with an existing advanced cache configuration:

1. Add `oi-distributed-rest-sessions` scheme to the existing `<caching-scheme-mapping>` section:

```
<cache-mapping>
 <cache-name>OpenIntRest_*</cache-name>
 <scheme-name>oi-distributed-rest-sessions</scheme-name>
</cache-mapping>
```

2. Add `<distributed-scheme>` to the existing `<caching-schemes>` section:

```
<distributed-scheme>
 <scheme-name>oi-distributed-rest-sessions</scheme-name>
 <service-name>DistributedCache</service-name>
 <backing-map-scheme>
 <read-write-backing-map-scheme>
 <internal-cache-scheme>
 <local-scheme>
 <high-units>0</high-units>
```

```
<expiry-delay>0</expiry-delay>
</local-scheme>
</internal-cache-scheme>
<listener>
<class-scheme>
<class-name>
com.siebel.openint.cache.OpenIntRestEvictionListener
</class-name>
<init-params>
<init-param>
<param-name>context</param-name>
<param-value>{manager-context}</param-value>
</init-param>
</init-params>
</class-scheme>
</listener>
</read-write-backing-map-scheme>
</backing-map-scheme>
<autostart>true</autostart>
</distributed-scheme>
```

3. Complete the cache configuration in SMC:
    a. Stop Open Integration.
    b. Open the existing Enterprise Cache Server profile in SMC.
    c. Under **Cache Configuration**, enable **Advanced**.
    d. Save a backup copy of the existing cache configuration XML.
    e. Add the 26.8 REST mapping and scheme.
    f. Make sure all the prerequisites for Advanced configuration are available. For more information, see *Prerequisites for Advanced Configuration*.
    g. Retain `ExtendTcpProxyService` and all valid customer-specific configuration.
    h. Do not duplicate cache mappings, scheme names, service names, listeners, or proxy definitions.
    i. Save the profile and redeploy the Enterprise Cache Server deployment using the standard SMC procedure.
    j. Start Open Integration after the deployment completes successfully.

Note: You must merge the 26.8 Open Integration configuration into the existing XML. Do not replace the existing cache XML.

Scenario 4: Upgrade with an Existing Non-Advanced Cache Configuration

You can use this scenario when Enterprise Cache was deployed by using the standard non-Advanced SMC configuration in the earlier release.

To upgrade with an existing non-advanced cache configuration:

1. Stop Open Integration.
2. Delete the existing Enterprise Cache Server deployment from SMC.
3. Create a new non-Advanced Enterprise Cache Server profile.
4. Configure the Enterprise Cache server profile. For more information, refer the *Introduction to Coherence Cache* documentation.
5. Deploy the new Enterprise Cache Server profile.
6. Confirm that the deployment completes successfully.
7. Start Open Integration.

No custom Open Integration cache XML is required.

Note: Do not manually edit generated Coherence XML for a non-Advanced profile. SMC generates and manages the required cache configuration.

Scenario 5: Upgrade Without Existing Cache Using a Non-Advanced Configuration

You can use this scenario when the earlier release did not use Enterprise Cache and you want the upgraded environment to use a non-Advanced configuration.

To upgrade with an existing cache using a non-advanced configuration:

1. Stop Open Integration, if it is running.
2. Create a new non-Advanced Enterprise Cache Server profile in SMC.
3. Configure the Enterprise Cache server profile. For more information, refer the *Introduction to Coherence Cache* documentation.
4. Deploy the Enterprise Cache Server profile.
5. Confirm that the deployment completes successfully.
6. Start Open Integration.

No custom Open Integration cache XML is required.

Scenario 6: Upgrade Without Existing Cache Using an Advanced Configuration

You can use this scenario when the earlier release did not use Enterprise Cache, and you want the upgraded Siebel CRM environment to use Open Integration cache with an Advanced configuration.

To upgrade without an existing cache using an Advanced configuration:

1. Stop Open Integration.
2. Create a new Enterprise Cache Server profile in SMC.
3. Under **Cache Configuration**, enable **Advanced**.
4. Configure Enterprise Cache installation and general settings. For more information, refer the *Introduction to Coherence Cache* documentation.
5. Make sure all the prerequisites for Advanced configuration are available. For more information, see *Prerequisites for Advanced Configuration*.
6. Deploy the Enterprise Cache Server profile.
7. Confirm that the deployment completes successfully.
8. Start Open Integration.

No XML merge is required because no earlier Enterprise Cache configuration exists.

Validating the Deployment

After you complete the applicable scenario:

1. Confirm in SMC that the Enterprise Cache Server deployment is running.
2. Confirm that `DistributedCache` and `DistributedConcurrentCache` start successfully.
3. Confirm that `ExtendTcpProxyService` and `OIExtendTcpProxyService` start successfully.
4. Review Enterprise Cache Server logs for cache-configuration, listener, cluster, and proxy-service errors.
5. Confirm that Open Integration starts successfully.

6. Confirm that Open Integration functions that require cache access work successfully.

Recovering from Coherence Request Timeout Errors

You can use this procedure when Open Integration reports a Coherence request timeout.

The following exception in the Open Integration logs indicates that Coherence did not respond to a request within the configured timeout period:

```
com.tangosol.net.RequestTimeoutException: Request timed out
```

A Coherence request timeout can cause inconsistent system behavior because Open Integration may be unable to read from or update the shared cache. To resolve this issue:

1. Stop Open Integration.
2. Restart Coherence by restarting the Enterprise Cache Server deployment.
3. Restart the Object Managers used by Open Integration.
4. Investigate the underlying Coherence network connectivity or latency issue.
5. If Coherence network latency is causing the timeout, configure an appropriate value for `Openint.cache.request-timeout`.
6. Start Open Integration.
7. Verify that Open Integration connects to Coherence successfully and that no new `RequestTimeoutException` errors are reported.

Note: Restarting only Open Integration is not sufficient. You must restart Coherence and the Object Managers used by Open Integration before you start Open Integration again.

Configuring the Open Integration Cache Request Timeout

The default value of `Openint.cache.request-timeout` is 2 seconds. If network latency between Open Integration and Coherence causes timeouts, increase the parameter gradually until the timeout is resolved. You must use the lowest request-timeout value that supports the observed Coherence network latency.

Note: Increasing `Openint.cache.request-timeout` accommodates longer Coherence response times. It does not resolve the underlying network issue. Continue to investigate and resolve the source of the latency.

Increasing `Openint.cache.request-timeout` can have the following effects:

- Requests wait longer for a Coherence response before timing out.
- Requests served by Open Integration can take longer to complete.
- More requests can remain active at the same time.
- Load on the Siebel Server can increase.
- The number of Open Integration sessions can increase beyond the expected level.

You must use the lowest request-timeout value that supports the observed Coherence network latency.

Advanced XML Reference

You can use the following XML as a reference for the required Open Integration Advanced configuration for Siebel CRM. This reference XML shows only the Open Integration additions. You must merge these entries into your existing `cache-config` file.

Note: If you already use an Advanced configuration, merge these Open Integration entries into your current XML. Retain all valid customer-specific configuration.

```
<cache-config xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
 xmlns="http://xmlns.oracle.com/coherence/coherence-cache-config"
 xsi:schemaLocation="http://xmlns.oracle.com/coherence/coherence-cache-config coherence-cache-config.xsd">

 <caching-scheme-mapping>
 <cache-mapping>
 <cache-name>OpenInt_*</cache-name>
 <scheme-name>oi-distributed-sessions</scheme-name>
 </cache-mapping>
 <cache-mapping>
 <cache-name>OpenIntRest_*</cache-name>
 <scheme-name>oi-distributed-rest-sessions</scheme-name>
 </cache-mapping>
 <cache-mapping>
 <cache-name>OpenIntAtomic_*_CurrentTasks</cache-name>
 <scheme-name>oi-concurrent</scheme-name>
 </cache-mapping>
 </caching-scheme-mapping>

 <caching-schemes>
 <distributed-scheme>
 <scheme-name>oi-distributed-sessions</scheme-name>
 <service-name>DistributedCache</service-name>
 <backing-map-scheme>
 <read-write-backing-map-scheme>
 <internal-cache-scheme>
 <local-scheme>
 <high-units>0</high-units>
 <expiry-delay>0</expiry-delay>
 </local-scheme>
 </internal-cache-scheme>
 <listener>
 <class-scheme>
 <class-name>
 com.siebel.openint.cache.OpenIntEvictionListener
 </class-name>
 <init-params>
 <init-param>
 <param-name>context</param-name>
 <param-value>{manager-context}</param-value>
 </init-param>
 </init-params>
 </class-scheme>
 </listener>
 </read-write-backing-map-scheme>
 </backing-map-scheme>
 <autostart>true</autostart>
 </distributed-scheme>

 <distributed-scheme>
 <scheme-name>oi-distributed-rest-sessions</scheme-name>
 <service-name>DistributedCache</service-name>
 <backing-map-scheme>
 <read-write-backing-map-scheme>
 <internal-cache-scheme>
 <local-scheme>
 <high-units>0</high-units>
 <expiry-delay>0</expiry-delay>
 </local-scheme>
 </internal-cache-scheme>
 <listener>
```

```
<class-scheme>
<class-name>
com.siebel.openint.cache.OpenIntRestEvictionListener
</class-name>
<init-params>
<init-param>
<param-name>context</param-name>
<param-value>{manager-context}</param-value>
</init-param>
</init-params>
</class-scheme>
</listener>
</read-write-backing-map-scheme>
</backing-map-scheme>
<autostart>true</autostart>
</distributed-scheme>

<distributed-scheme>
<scheme-name>oi-concurrent</scheme-name>
<service-name>DistributedConcurrentCache</service-name>
<backing-map-scheme>
<partitioned>true</partitioned>
<local-scheme>
<unit-calculator>BINARY</unit-calculator>
</local-scheme>
</backing-map-scheme>
<autostart>true</autostart>
</distributed-scheme>

<proxy-scheme>
<scheme-name>OIExtendTcpProxyService</scheme-name>
<service-name>OIExtendTcpProxyService</service-name>
<acceptor-config>
<tcp-acceptor>
<socket-provider system-property="coherence.extend.socketprovider"/>
</tcp-acceptor>
</acceptor-config>
<autostart>true</autostart>
</proxy-scheme>
</caching-schemes>
</cache-config>
```

Siebel AI Connectors

This topic provides an overview of Siebel AI Connectors. Siebel AI Connectors enable customers to expose selected Siebel Open Integration REST APIs as Model Context Protocol (MCP)-compatible tools and servers for consumption by AI agents and assistant applications.

Siebel Open Integration exposes Siebel objects, services and workflows through REST APIs. These APIs are powerful and flexible for direct system-to-system integration. However, AI agents require a more structured interface for interacting with enterprise systems.

Siebel AI Connectors provide a configuration-driven framework for exposing selected Open Integration REST APIs as MCP tools. Customers identify the Open Integration REST APIs they want to expose, define MCP tool metadata & maintain a centralized catalog, group those tools into MCP Servers, and once configured deploy these servers either on-premises or via Siebel Cloud Manager(SCM). The Siebel AI Connectors runtime then realizes the configured MCP Servers and exposes runtime endpoints that AI agents can use to discover and invoke the configured tools.

By using Siebel AI Connectors, customers can expose only the Open Integration operations that are relevant for an AI-agent use case, keep access to Siebel data and actions controlled, and align agent behavior with business processes.

This topic covers the following sections:

1. *Installing Siebel AI Connectors*
2. *Configuring Siebel AI Connectors*
3. *Deploying Siebel AI Connectors*

Installing Siebel AI Connectors

This topic describes how to install Siebel AI Connectors.

Siebel AI Connectors can work with on-premises environments and in Kubernetes-based environments managed through Siebel Cloud Manager. The deployment model determines how the runtime package and configuration artifacts are delivered, but the core runtime behavior remains the same.

In an on-premises deployment, the runtime package includes the artifacts required to run Siebel AI Connectors in the target environment.

In a Siebel Cloud Manager deployment, the Siebel AI Connectors runtime is deployed in a Kubernetes-based environment. Configuration artifacts are provided as part of the deployment process. More details to how we can work with Siebel Cloud Manager to deploy and manage Siebel AI Connectors can be found in the Deploying Siebel AI Connectors section of this topic. For more details on SCM, please read the Administration Bookshelf guide for Siebel Cloud Manager.

Note:

- Open Integration installation and configuration is a prerequisite to ensure that your Siebel AI Connectors works properly.
- During the installation process, don't modify the runtime JAR or dependent libraries. Update only the configuration files.

On-Premises Package Contents

The on-premises runtime package includes the runtime JAR, dependent libraries, and configuration template files.

The package includes the following artifacts:

```
mcp-runtime/
  ├── mcp-runtime-<version>.jar. # Do not Modify
  ├── libs/ # Do not Modify
  ├── application.yaml # Configuration File
  ├── logging.properties # Configuration File
  ├── MCP_CONFIG.json # Configuration Files
  ├── FINAL_SPECIFICATION.json # Configuration Files
  ├── OI_CONFIG.json # Configuration Files
```

On-Premises Installation for Windows & Linux

Windows Installer

Interactive Installation

Use this procedure to install Siebel AI Connectors on Windows by using the interactive installer.

1. Download and extract the Siebel AI Connectors installer artifacts.

2. Review the README file provided with the installer package.
3. Start the installer by running the Windows installer command.
4. Select the installation directory.
5. Select the Siebel AI Connectors component.
6. Review the installation summary.
7. Click **Install**.
8. After installation completes, verify that the Siebel AI Connectors artifacts are created in the selected installation directory.
9. Configure the runtime artifacts before starting the Siebel MCP Server.

Silent Installation

Use this procedure to install Siebel AI Connectors on Windows in silent mode.

1. Run the installer and proceed to the Summary screen.
2. Save the response file.
3. Review the response file and verify that the installation directory is correct.
4. Run the installer in silent mode using the generated response file.
5. After installation completes, verify that the Siebel AI Connectors artifacts are created in the selected installation directory.

Linux Installer

Interactive Installation

Use this procedure to install Siebel AI Connectors on Linux by using the interactive installer.

1. Download and extract the Siebel AI Connectors installer artifacts.
2. Review the README file provided with the installer package.
3. Start the installer by running the Linux installer command.
4. Select the installation directory.
5. Select the Siebel AI Connectors component.
6. Review the installation summary.
7. Click **Install**.
8. After installation completes, verify that the Siebel AI Connectors artifacts are created in the selected installation directory.
9. Configure the runtime artifacts before starting the Siebel MCP Server.

Silent Installation

Use this procedure to install Siebel AI Connectors on Linux in silent mode.

1. Run the installer and proceed to the Summary screen.
2. Save the response file.
3. Review the response file and verify that the installation directory is correct.
4. Run the installer in silent mode using the generated response file.
5. After installation completes, verify that the Siebel AI Connectors artifacts are created in the selected installation directory.

Configuring Siebel AI Connectors

This section describes how to configure Siebel AI Connectors, prepare MCP tools from Siebel Open Integration REST APIs, secure MCP endpoints, and validate runtime behavior.

Siebel AI Connectors are intended to be used with Siebel Open Integration. After Open Integration is configured and running, it exposes the resources defined for API generation under `config.json` as standard REST APIs. The

corresponding OpenAPI specification available from the Open Integration runtime describes the exposed resources, paths, HTTP methods, parameters, request bodies, and schemas.

Customers use the Open Integration REST APIs and the corresponding OpenAPI specification as the starting point to decide which operations should be selectively exposed as MCP tools. The selected operations and tool metadata are represented in `FINAL_SPECIFICATION.json`, grouped into MCP Servers using `MCP_CONFIG.json`, connected to Open Integration using `OI_CONFIG.json`, and loaded by the runtime using `application.yaml` and `logging.properties`.

Configuration Prerequisites

Before configuring or deploying Siebel AI Connectors, complete the following prerequisites.

Open Integration

Siebel Open Integration must be installed, configured, and running before Siebel AI Connectors can be configured.

Check	Description
Open Integration is installed and configured.	Siebel AI Connectors depend on Open Integration REST APIs as the source APIs for MCP tools.
Open Integration runtime is running.	The runtime must expose the configured resources as standard REST APIs.
OpenAPI specification is available from the Open Integration runtime.	The OpenAPI specification describes the resources, paths, HTTP methods, parameters, request bodies, and schemas available for tool creation.
Target Open Integration server details are known.	These values are required in `OI_CONFIG.json`, including server name, scheme, host, port, and context.

Tool Selection

Before creating or updating `FINAL_SPECIFICATION.json`, identify which Open Integration operations should be exposed as MCP tools.

Check	Description
Required business use case is identified.	Only expose APIs required by the intended AI-agent use case.
Required Open Integration resources are identified.	For example, Account, Contact, Product, Service Request, or other configured resources.
Required API operations are selected.	Select the required paths and HTTP methods, such as `get`, `post`, `put`, or `delete`.
Tool names and descriptions are planned.	Tool names should be stable, and descriptions should be clear enough for MCP clients and AI agents.
Required request inputs are understood.	Review path, query, header parameters, and request body schemas.
Security impact is reviewed.	Avoid exposing create, update, or delete operations unless the AI-agent use case requires them.

Runtime Package

The Siebel AI Connectors runtime package must be available on the target environment.

Check	Description
`mcp-runtime-<version>.jar` is present.	Runtime application binary. Do not modify.
`libs/` is present.	Runtime dependencies. Do not modify.
`application.yaml` is present.	Defines runtime server settings, security settings, and configuration file locations.
`logging.properties` is present.	Defines runtime logging behavior.
`MCP_CONFIG.json` is present.	Defines MCP Servers and tool mappings.
`FINAL_SPECIFICATION.json` is present.	Defines selected tools in OpenAPI-compatible format.
`OI_CONFIG.json` is present.	Defines Open Integration server instances.

Security

If JWT authentication is enabled for Siebel AI Connectors, verify the following:

Check	Description
Identity provider is configured.	For example, OCI Identity Domain or the configured OAuth provider.
JWT issuer is known.	Required in **`security.jwt.issuer`**.
JWT audience is known.	Required in **`security.jwt.audience`**.
Required OAuth scope is known.	Required in **`security.jwt.required-scope`**. Scope Siebel AI Connectors should match with what is defined for Open Integration.
JWKS URI is available.	Required in **`security.jwt.jwks-uri`**.
Open Integration authentication is aligned.	If Siebel AI Connectors uses OAuth/JWT authentication, Open Integration must also be configured to accept the corresponding authentication flow.
Test token can be generated.	Required for secured endpoint validation.

High-Level Configuration Overview

To ensure that you can start your MCP Servers using Siebel AI Connectors, the following diagram provides a high level guidance on the activities you need to perform when working with the product:

1. **Review Open Integration API(s):** Open Integration is the source of all MCP tools available in **`FINAL_SPECIFICATION.json`** for Siebel AI Connectors. Begin by reviewing the OpenAPI specification generated by Open Integration, for the target resource & identify the REST operations that support your AI use case. Rather than exposing every available operation, select only the APIs that an AI agent needs to perform its tasks. This helps minimize unnecessary access and simplifies tool management.
2. **Define MCP Tool(s):** After identifying the required operations, move relevant APIs into the **`FINAL_SPECIFICATION.json`** and create MCP tool definitions that describe how those APIs will be presented to AI agents. Each MCP tool represents a specific business capability, such as retrieving account information or updating a service request. The tool definitions become the catalog of capabilities that can later be assigned to MCP Servers.
3. **Add Open Integration instance(s):** Ensure that the **`OI_CONFIG.json`** is updated with relevant Open Integration instances which have relevant APIs updated as part of the **`FINAL_SPECIFICATION.json`** file.

4. **Organize Tools into MCP Server(s):** Group related MCP tools into one or more MCP Servers in the `MCP_CONFIG.json` file. A server can represent a business domain, application area, or functional capability. For example, one server might expose customer-related tools while another exposes product-related operations. Organizing tools into logical servers makes them easier for AI agents to discover and consume.
5. **Configure the Runtime:** Once the tools and servers have been defined, configure the remaining configuration files - `application.yaml` & `logging.properties` with appropriate details. The runtime uses these configured files to establish connections with the configured Open Integration instances, initializes the MCP Servers, and exposes the corresponding endpoints for AI clients.
6. **Deploy and Validate:** Deploy the configured runtime to your target environment, either on-premises or using Siebel Cloud Manager. After deployment, verify that:
    - The runtime starts successfully.
    - The MCP Server endpoints are available.
    - AI clients can discover the configured servers.
    - The exposed tools successfully invoke the corresponding Open Integration REST APIs.

Although creating an MCP Server is primarily a configuration activity, the runtime uses several configuration artifacts to describe the tools, servers, runtime settings, Open Integration connections, and logging behavior. These artifacts work together to define what capabilities are exposed and how the runtime operates. Each artifact has a specific purpose and should be configured consistently before deployment.

Inspecting Open Integration

Open Integration is the source for APIs exposed through Siebel AI Connectors. After Open Integration is configured and running, the resources defined for API generation under `config.json` are exposed as standard REST APIs. The corresponding OpenAPI specification from the Open Integration runtime provides metadata for the exposed resources, including paths, HTTP methods, parameters, request bodies, and schemas.

Use these available Open Integration APIs as the starting point for tool selection. Not every Open Integration operation needs to be exposed as an MCP tool. Select only the operations required for the AI-agent use case.

To prepare MCP tools:

1. Confirm that the required Open Integration resources are configured for API generation under `config.json`.
2. Start or verify the Open Integration runtime.
3. Use the `describe` operation available for the target Siebel artifact configured on Open Integration, to fetch and review the corresponding OpenAPI specification.
4. Identify the API paths and HTTP methods that should be exposed to AI agents.
5. Define the selected operations and MCP tool metadata in `FINAL_SPECIFICATION.json`.
6. Group the selected tools into MCP Servers using `MCP_CONFIG.json`.

The following example shows an Open Integration Account resource exposed as REST API operations:

```
{
 "paths": {
 "/openintegration/v1.0/data/Account/Account/": {
 "get": {
 "operationId": "openintegration_v1.0_data_Account_Account__get"
 },
 "put": {
 "operationId": "openintegration_v1.0_data_Account_Account__put"
 },
 "post": {
 "operationId": "openintegration_v1.0_data_Account_Account__post"
 }
 }
 }
```

```
}
```

In this example, the Account resource exposes `get`, `put`, and `post` operations. The implementation team can evaluate these operations and decide which ones should be selectively created as MCP tools.

Open Integration Operation	Expose as MCP Tool?	Guidance
GET /openintegration/v1.0/data/Account/Account/	Common	Expose when the AI-agent use case requires account lookup or retrieval.
PUT /openintegration/v1.0/data/Account/Account/	Conditional	Expose only if the AI-agent use case is allowed to update account data.
POST /openintegration/v1.0/data/Account/Account/	Conditional	Expose only if the AI-agent use case is allowed to create account data.

After the tool selection is complete, define the selected operations in `FINAL_SPECIFICATION.json` and map them to MCP Server endpoints using `MCP_CONFIG.json`.

Configuring Siebel AI Connectors Files

Only the configuration files should be updated. Do not modify `mcp-runtime-<version>.jar` or files under `libs/`. The below table captures details on individual configuration files available for developers to make necessary changes to configure MCP Tools & Servers:

File or Folder	Purpose	Update Guidance
FINAL_SPECIFICATION.json	Cumulated tools in OpenAPI-compatible format. Defines MCP tool metadata and Open Integration operation mappings.	Generated or updated by configuration-time tooling. Do not manually update unless instructed.
MCP_CONFIG.json	MCP Server configuration. Maps MCP Servers to tool names from `FINAL_SPECIFICATION.json`.	Update when MCP Server grouping or endpoint mapping changes.
application.yaml	Runtime configuration. Defines server host, port, security settings, and paths to runtime configuration files.	Update when runtime settings, security settings, or file locations change.
OI_CONFIG.json	Open Integration instance configuration. Defines Open Integration servers used by the runtime.	Update when Open Integration server details change.
logging.properties	Runtime logging configuration.	Update when logging levels or logging behavior need to change.

FINAL_SPECIFICATION.json

`FINAL_SPECIFICATION.json` is the runtime-consumable tool specification used by Siebel AI Connectors to create MCP tool definitions from selected Open Integration REST API operations. It is modeled on the OpenAPI Specification and allows developers to define:

- Open Integration API paths and HTTP methods selected for exposure.
- List of tool names available as part of the FINAL_SPECIFICATION.json
- Tool metadata - tool name, descriptions and so on are defined for selected Open Integration paths and HTTP operations.

- Open Integration server mappings.
- Parameters, request bodies, and schemas used to build and validate MCP tool inputs.

The delivered template uses the following top-level structure:

```
{
 "openapi": "<version>",
 "info": {
 "title": "<Title>",
 "description": "<Description>",
 "contact": {
 "email": "sample@sample.com"
 },
 "version": "<version>"
 },
 "externalDocs": {
 "description": "<description>",
 "url": "<url>"
 },
 "servers": [
 {
 "url": "<Server Base Url>",
 "x-server-name": "<Server Instance Name>"
 }
 ],
 "x-tools": [
 "<Tool Name>"
 ],
 "tags": [
 {
 "name": "<tag>"
 }
 ],
 "paths": {
 "<Path>": {
 "<http-method>": {
 "x-tool-name": "<Tool Name>",
 "x-server-name": "<Server Instance Name>",
 "tags": [
 "<tag>"
 ],
 "operationId": "<Operation Id>",
 "description": "<Description>",
 "summary": "<Summary>",
 "parameters": [],
 "requestBody": {},
 "responses": {}
 }
 }
 },
 "components": {
 "schemas": {}
 },
 "x-original-swagger-version": "3.0"
}
```

Top-Level Fields

Field	Required	Description
components	Recommended	Contains schemas referenced by parameters or request bodies.

Field	Required	Description
externalDocs	Optional	External documentation reference.
info	Recommended	OpenAPI metadata, including title, description, contact, and version.
openapi	Yes	OpenAPI version used by the specification.
paths	Yes	Contains Open Integration API paths and operations selected for tool generation.
servers	Yes	Server entries from the OpenAPI specification. Runtime reads these for server-name and URL comparison or logging. The downstream base URL is resolved from `OI_CONFIG.json`.
tags	Optional	OpenAPI tags used for grouping operations.
x-original-swagger-version	Optional	Preserves source Swagger/OpenAPI version metadata.
x-tools	Yes	List of all tools present in the file.

Additional Details on Fields for FINAL_SPECIFICATION.json

The following describes additional information on the different fields described in the above table, in scope of the FINAL_SPECIFICATION.json

1. servers

   The `servers` array contains OpenAPI server definitions and logical server names.

```
"servers": [
 {
 "url": "<Server Base Url>",
 "x-server-name": "<Server Instance Name>"
 }
]
```

Field	Required	Description
url	Yes	Server base URL from the OpenAPI specification.
x-server-name	Yes	Logical server instance name. Operation-level `x-server-name` values use this name to identify the Open Integration server mapping.

   The runtime reads top-level `servers` for server-name and URL comparison or logging. It does not use the top-level server URL directly as the downstream base URL. The downstream server URL is resolved from `OI_CONFIG.json`.

2. x-tools

   The `x-tools` field lists all tool names present in `FINAL_SPECIFICATION.json`. Each operation-level `x-tool-name` should be present in the top-level `x-tools` array.

```
"x-tools": [
 "Account GET",
 "Account PUT",
 "Account POST"
]
```

3. paths

   The `paths` object contains the selected Open Integration API paths and HTTP method that can be realized as MCP Tools by the Siebel AI Connectors runtime.

Sample:

```
"paths": {
 "/openintegration/v1.0/data/Account/Account/": {
 "get": {
 .....
 },
 "put": {
 .....
 }
 }
}
```

Field	Required	Description
Path key	Yes	OpenAPI path, for example `/accounts/{id}` or `/openintegration/v1.0/data/Account/Account/`.
HTTP method	Yes	HTTP operation under the path, such as `get`, `post`, `put`, or `delete`.

4. HTTP method Fields

   For Siebel AI Connectors, we map individual HTTP methods(GET, PUT, POST and DELETE) defined under a "`path`" to an MCP Tool in the FINAL_SPECIFICATION.json file. Each selected HTTP method defines the OpenAPI operation and MCP-specific metadata. The following table captures relevant information on the different key-value pairs that should be configured:

Field	Required	Description
x-tool-name	Recommended	Primary source for the MCP tool name. If absent or blank, `operationId` is used as the fallback tool name.
x-server-name	Optional	Identifies which server entry from `OI_CONFIG.json` the tool sends the downstream request to. If missing, the

Field	Required	Description
		default server from `OI_CONFIG.json` is used.
operationId	Yes	Used as fallback MCP tool name when `x-tool-name` is absent or blank. Also used in description fallback behavior.
description	Recommended	Primary source for the MCP tool description. Can be updated by user to reflect the tool description they would want to provide for the selected operation.
summary	Optional	Fallback MCP tool description when `description` is absent or blank.
tags	Optional	Provided by the Open Integration OpenAPI Spec.
parameters	Conditional	Provided by the Open Integration OpenAPI Spec.
requestBody	Conditional	Provided by the Open Integration OpenAPI Spec. Required when the operation expects a request body. Used to build the body argument in the MCP input schema.
responses	Recommended	Provided by the Open Integration OpenAPI Spec.

5. parameters

Parameters are available for certain HTTP methods available for a given `path`. These parameters are specified in Open Integration's OpenAPI spec and are added as needed in the `FINAL_SPECIFICATION.json` for the associated path and HTTP method.

Sample:

```
"parameters": [
 {
 "name": "Id",
 "in": "query",
 "description": "Account identifier.",
 "schema": {
 "type": "string"
 }
 }
]
```

Field	Required	Description
in	Yes	Provided via the Open Integration OpenAPI Spec

Field	Required	Description
name	Yes	Provided via the Open Integration OpenAPI Spec
schema	Yes	Provided via the Open Integration OpenAPI Spec
description	Recommended	Optionally available. Should be added for the parameter for proper usage with the MCP tool.

6. components.schemas

   The `components.schemas` section defines schemas used by parameters and request bodies.

   Sample:

```
"components": {
 "schemas": {
 "AccountUpdate": {
 "type": "object",
 "additionalProperties": false,
 "required": [
 "Name"
 ],
 "properties": {
 "Name": {
 "type": "string",
 "description": "Account name."
 }
 }
 }
 }
}
```

Field	Description
required	Validated for request body. This is distinct from the generated top-level tool required array.
additionalProperties	Controls whether unexpected body fields are allowed in a request body. Can be set to true or false. If set to `false`, unexpected body fields are rejected. If set to true, additional fields are allowed as part of the Request Body.
description	Exposed to clients. Body descriptions appear on `properties.body`.
discriminator	Preserved. Runtime has no discriminator-specific logic, although schema composition may apply where supported.

MCP_CONFIG.json

`MCP_CONFIG.json` defines MCP Servers and maps each MCP Server to tool names from `FINAL_SPECIFICATION.json`.

Sample:

```
{
 "mcp-servers": [
 {
 "mcp-server-name": "GET",
 "mcp-server-path": "/account1",
 "mcp-server-description": "Account 1 Description",
 "mcp-server-version": "1.0.0",
 "mcp-server-tools": [
 "Account GET",
 "Account PUT",
 "Account POST"
 ]
 },
 {
 "mcp-server-name": "PUT",
 "mcp-server-path": "/account2",
 "mcp-server-description": "Account 2 Description",
 "mcp-server-version": "1.0.0",
 "mcp-server-tools": [
 "Account GET",
 "AccountMethod POST",
 "AccountKeyMethod POST"
 ]
 }
 ]
}
```

Field	Required	Description
mcp-servers	Yes	Array of MCP Server definitions.
mcp-server-name	Yes	Name of the MCP Server.
mcp-server-path	Yes	Runtime endpoint path for the MCP Server. Must be unique for each MCP Server.
mcp-server-description	Yes	Description of the MCP Server.
mcp-server-version	Yes	MCP Server version.
mcp-server-tools	Yes	List of tool names exposed by this MCP Server. Each value should be present under x-tools in `FINAL_SPECIFICATION.json`

Rules:

- Each `mcp-server-path` must be unique.
- Each tool in `mcp-server-tools` must exist in `FINAL_SPECIFICATION.json`.
- Tool names are matched by exact string.

OI_CONFIG.json

`OI_CONFIG.json` defines Open Integration instance(s) used by the runtime.

Sample:

```
{
 "default-server": "<server-name-1>",
 "servers": [
 {
 "name": "<server-name-1>",
 "scheme": "<scheme>",
 "host": "<host>",
 "port": 8433,
 "context": "<context>"
 },
 {
 "name": "<server-name-2>",
 "scheme": "<scheme>",
 "host": "<host>",
 "port": 8433,
 "context": "<context>"
 }
 ]
}
```

Field	Required	Description
default-server	Yes	Default Open Integration server. Used when operation-level **x-server-name** is missing or unknown.
servers	Yes	List of Open Integration server definitions.
name	Yes	Logical Open Integration server name. This should match operation-level **x-server-name** values in **FINAL_SPECIFICATION.json**.
scheme	Yes	Protocol used for Open Integration, such as **http** or **https**.
host	Yes	Open Integration host.
port	Yes	Open Integration port.
context	Yes	Open Integration context path.

Rules:

- **default-server** must match one of the **servers[].name** values.
- Each **servers[].name** must be unique.
- Operation-level **x-server-name** values in **FINAL_SPECIFICATION.json** should resolve to entries in **OI_CONFIG.json**.
- If operation-level **x-server-name** is missing or unknown, the runtime uses **default-server**.

logging.properties

The **logging.properties** defines runtime logging behavior and must be referenced from **application.yaml** using **mcp.logging-properties-file**.

The file should be placed in the same folder as **mcp-runtime-<version>.jar** unless the path in **application.yaml** is updated.

```
handlers=io.helidon.logging.jul.HelidonConsoleHandler
java.util.logging.SimpleFormatter.format=%1$tY.%1$tm.%1$td %1$tH:%1$tM:%1$tS.%1$tL %4$s %5$s%6$s%n
```

```
# Global logging level. Can be overridden by specific loggers
.level=SEVERE
io.helidon.level=SEVERE
```

Field	Description
handlers	Defines the logging handler used by the runtime.
java.util.logging.SimpleFormatter.format	Defines the log message format.
.level	Global logging level. Can be overridden by specific loggers.
io.helidon.level	Logging level for Helidon runtime loggers.

application.yaml

`application.yaml` defines runtime server settings, JWT security settings, and configuration file paths.

```
Sample:

server:
 port: <port>
 host: <host>

security:
 jwt:
 enabled: true
 issuer: <issuer>
 audience: <audience>
 required-scope: <scope>
 clock-skew-seconds: <seconds>
 jwks-uri: "<jwks-uri>"
 jwks-cache-duration-seconds: <duration>

mcp:
 server-config-file: <path to MCP_CONFIG.json>
 final-spec-file: <path to FINAL_SPECIFICATION.json>
 logging-properties-file: <path to logging.properties>
 open-int-config-file: <path to OI_CONFIG.json>
```

Field	Required	Description
server.port	Yes	Runtime server port. Change this if the configured port is already in use.
server.host	Yes	Runtime server host.
security.jwt.enabled	Required when JWT is used	Enables or disables JWT validation.
security.jwt.issuer	Required when JWT is used	Expected token issuer.
security.jwt.audience	Required when JWT is used	Expected token audience.
security.jwt.required-scope	Required when JWT is used	Required OAuth scope for MCP requests.
security.jwt.clock-skew-seconds	Recommended when JWT is used	Allowed clock skew during token validation.
security.jwt.jwks-uri	Required when JWT is used	JWKS URI used to validate JWT signatures.
security.jwt.jwks-cache-duration-seconds	Recommended when JWT is used	Duration for JWKS cache.

Field	Required	Description
mcp.server-config-file	Yes	Path to `MCP_CONFIG.json`.
mcp.final-spec-file	Yes	Path to `FINAL_SPECIFICATION.json`.
mcp.logging-properties-file	Yes	Path to `logging.properties`.
mcp.open-int-config-file	Yes	Path to `OI_CONFIG.json`.

Security and Authentication

Siebel AI Connectors support OAuth 2.0 JWT-based authentication. When JWT authentication is enabled, each request to an MCP Server endpoint must include a valid access token.

The runtime validates the token using the JWT configuration in `application.yaml`.

```
security:
 jwt:
 enabled: true
 issuer: <issuer>
 audience: <audience>
 required-scope: <scope>
 clock-skew-seconds: <seconds>
 jwks-uri: "<jwks-uri>"
 jwks-cache-duration-seconds: <duration>
```

Field	Required	Description
security.jwt.enabled	Yes	Enables JWT authentication.
security.jwt.issuer	Yes	Expected token issuer.
security.jwt.audience	Yes	Expected token audience.
security.jwt.required-scope	Yes	OAuth scope required to invoke MCP endpoints.
security.jwt.clock-skew-seconds	Recommended	Allowed clock skew during token validation.
security.jwt.jwks-uri	Yes	JWKS endpoint used to validate token signatures.
security.jwt.jwks-cache-duration-seconds	Recommended	Duration for JWKS cache.

In a secured deployment:

1. The MCP client generates an OAuth access token from the configured identity provider.
2. The MCP client sends the request to the MCP Server endpoint with an authorization header.
3. The Siebel AI Connectors runtime validates the JWT.
4. After validation, the runtime invokes the configured Open Integration REST API.

Note: Authorization: Bearer <access_token>

The authentication configuration must be consistent across Siebel AI Connectors and Open Integration. If Siebel AI Connectors uses OAuth/JWT authentication, Open Integration must also be configured to accept the corresponding authentication flow.

Consistency Checklist

Before starting or deploying the runtime, verify the following:

Check	Required
Every tool in `MCP_CONFIG.json` has a mapping in `FINAL_SPECIFICATION.json`.	Yes
Every operation-level `x-tool-name` is listed in top-level `x-tools`.	Yes
`mcp.server-config-file` points to `MCP_CONFIG.json`.	Yes
`mcp.final-spec-file` points to `FINAL_SPECIFICATION.json`.	Yes
`mcp.logging-properties-file` points to `logging.properties`.	Yes
`mcp.open-int-config-file` points to `OI_CONFIG.json`.	Yes
Every operation-level `x-server-name` resolves to an entry in `OI_CONFIG.json`, or `default-server` is configured for fallback.	Yes
`OI_CONFIG.json` defines a valid `default-server`.	Yes
Each `mcp-server-path` is unique.	Yes
Request bodies use `application/json`.	Yes
Supported parameter locations are `path`, `query`, and `header`.	Yes

Deploying Siebel AI Connectors

This section describes how to deploy Siebel AI Connectors after the required configuration files are prepared.

Siebel AI Connectors can be deployed in an on-premises environment or through Siebel Cloud Manager. In both deployment models, the runtime uses the configured artifacts to create MCP Servers, expose MCP-compatible tools, and connect those tools to the selected Siebel Open Integration REST APIs.

Before deploying Siebel AI Connectors, verify that `application.yaml`, `MCP_CONFIG.json`, `FINAL_SPECIFICATION.json`, `OI_CONFIG.json`, and `logging.properties` are updated for the target environment. Also verify that Siebel Open Integration is installed, configured, and running.

The deployment topics include:

- Deploying Siebel AI Connectors - On-Premise
- Deploying Siebel AI Connectors - Siebel Cloud Manager
- Runtime Behavior and Validations

Deployment Prerequisites

Java and Certificate Prerequisites

For on-premises runtime startup, verify the following:

Prerequisite	Description
JRE 21 is available.	Configure **JAVA_HOME** and **PATH** for the JRE delivered or supported with the Siebel installer.
Required certificates are trusted.	If Open Integration uses TLS, import the required Siebel or Open Integration certificate into the Java truststore used by the runtime.
Truststore path and password are known.	Required when configuring Java truststore behavior for deployment or runtime.

Example certificate import command on Linux:

```
keytool -importcert -trustcacerts -alias siebel \
 -file "<FS path>/siebel.crt" \
 -keystore "<FS path>/jre/lib/security/cacerts" \
 -storepass changeit \
 -noprompt
```

Example certificate import command on Windows:

```
keytool -importcert -trustcacerts -alias siebel ^
 -file "<FS path>\siebel.crt" ^
 -keystore "<FS path>\jre\lib\security\cacerts" ^
 -storepass changeit ^
 -noprompt
```

SCM Deployment Prerequisites

For Siebel Cloud Manager deployment, verify the following before deploying Siebel AI Connectors:

Prerequisite	Description
Siebel Cloud Manager is available.	SCM APIs are used to create infrastructure and deploy Siebel AI Connectors.
Kubernetes environment is available.	For example, BYO OCNE, OKE, or another supported Kubernetes target.
Infrastructure input is available.	Namespace, kubeconfig path or cluster details, operator options, registry details, and GitOps details are required.
Customer Git repository is available.	Runtime configuration artifacts are maintained in Git.
Git access credentials are available.	HTTPS token, self-signed CA path, or SSH private key path may be required.
Git repository is registered with SCM.	SCM returns a **git_id** used in the deployment payload.
Container registry access is available.	Registry URL, user, password or token, and registry prefix are required.
Configuration files are committed to Git.	**application.yaml**, **MCP_CONFIG.json**, **FINAL_SPECIFICATION.json**, **OI_CONFIG.json**, and **logging.properties** must be available in the configured Git folder.
Truststore file is available when required.	The deployment payload can reference the Java truststore path and password.
Namespace is created when using BYO namespace.	If **byo_ns** is **true**, create the namespace before deployment and use the namespace value in the deployment **name** field.

Prerequisite	Description
Security context values are known when required.	If a custom security context is used, provide `run_as_user`, `run_as_group`, and `fs_group`.

Deploying Siebel AI Connectors - On-Premises

This topic describes how to deploy Siebel AI Connectors in an on-premises environment after the required configuration files are prepared.

In an on-premises deployment, Siebel AI Connectors are started from the delivered runtime package. Before starting the runtime, verify that the configuration files are present, that the Open Integration server details are correct, and that the Java prerequisites are complete.

Before You Begin

Before deploying Siebel AI Connectors on-premise, make sure that:

- Siebel Open Integration is installed, configured, and running.
- The Open Integration REST APIs selected for MCP tool exposure are available.
- The Siebel AI Connectors runtime package is installed.
- The required configuration files are updated for the target environment.
  - Navigate to the Siebel AI Connectors runtime directory and verify that the necessary files, libs and folders are present.
  - Verify that `application.yaml` contains the correct runtime settings and file paths.
    - `server.port` is available on the target host.
    - `server.host` is correct for the deployment.
    - The paths under `mcp` point to the correct files.
    - JWT settings are correct, if authentication is enabled.
  - Verify that `OI_CONFIG.json` contains the Open Integration server instances used by the runtime.
    - `default-server` matches one of the server names in the `servers` array.
    - Each server name is unique.
    - `scheme`, `host`, `port`, and `context` are correct for the Open Integration environment.
    - The runtime host can connect to each configured Open Integration server.
  - Verify that `FINAL_SPECIFICATION.json` contains the selected Open Integration operations and MCP tool metadata.
    - The required tools are present.
    - Tool names are unique.
    - Tool names used in `MCP_CONFIG.json` match the names in `FINAL_SPECIFICATION.json`.
    - The Open Integration server mappings are correct.
  - Verify that `MCP_CONFIG.json` defines the MCP Servers and maps each server to one or more tools from `FINAL_SPECIFICATION.json`.
    - Each `mcp-server-path` is unique.
    - Each tool listed in `mcp-server-tools` exists in `FINAL_SPECIFICATION.json`.
    - Tool names match exactly between `MCP_CONFIG.json` and `FINAL_SPECIFICATION.json`.
- JRE 21 is configured.
- Required certificates are imported into the Java truststore, if Open Integration uses TLS.

Start the Siebel AI Connectors Runtime

After verifying the configuration files and necessary prerequisites, start the runtime from the runtime directory.

For Windows, run:

```
java -cp ".\mcp-runtime-<version>.jar;.\libs\*" com.oracle.siebel.mcp.Main
```

For Linux, run:

```
java -cp "./mcp-runtime-<version>.jar:./libs/*" com.oracle.siebel.mcp.Main
```

Replace `<version>` with the version of the runtime JAR delivered in the package.

Verify Runtime Startup

After starting the runtime, review the startup logs.

Look for log messages that confirm the following:

- `FINAL_SPECIFICATION.json` was parsed successfully.
- Open Integration servers were loaded.
- The default Open Integration server was identified.
- Tools were processed from the OpenAPI specification.
- The runtime selected the Open Integration server for each tool.
- MCP tools were adapted from the OpenAPI specification.
- MCP Servers were started successfully.
- The base MCP Server URL was created.

Example log messages:

```
INFO Successfully parsed OpenAPI spec file:<Path>/FINAL_SPECIFICATION.json

INFO Loaded Open Integration servers: [OI123, OI456]; default server: OI456

INFO Processing Tool : [Account GET] , Method : [GET] , Path: [/openintegration/v1.0/data/Account/Account/]

INFO Using Open Integration server [OI456] for tool [Account GET]

INFO Adapted [20] tools from OpenAPI spec

INFO Started all channels in 8 milliseconds. 590 milliseconds since JVM startup. Java 21.0.11

INFO MCP Servers started with base url: 0.0.0.0:8081/siebelAIConnectors/mcp
```

If the runtime does not start successfully, review the logs and verify the configuration file paths, Open Integration server details, Java configuration, certificate configuration, and MCP tool mappings.

Verify MCP Server Access

After the Siebel AI Connectors runtime starts successfully, verify that the configured MCP Servers are available.

The runtime exposes MCP Servers using the base MCP path and the MCP Server path configured in `MCP_CONFIG.json`.

The MCP Server endpoint uses the following format:

```
<scheme>://<host>:<port>/siebelAIConnectors/mcp/<mcp-server-path>
```

Where:

Value	Description
<scheme>	The protocol used to access the Siebel AI Connectors runtime, such as **http** or **https**.
<host>	The host where the Siebel AI Connectors runtime is running.
<port>	The runtime port configured in **application.yaml**.
/siebelAIConnectors/mcp	The base path used by the Siebel AI Connectors runtime for MCP Servers.
<mcp-server-path>	The MCP Server path configured in **MCP_CONFIG.json**.

For example, if **application.yaml** configures the runtime on port **8081**, and **MCP_CONFIG.json** defines the following MCP Server path:

```
{
 "mcp-server-name": "Account MCP Server",
 "mcp-server-path": "/account",
 "mcp-server-description": "MCP Server for account-related Siebel operations.",
 "mcp-server-version": "1.0.0",
 "mcp-server-tools": [
 "get_account_details",
 "update_account_details"
 ]
}
```

Then the MCP Server endpoint is available at:

```
http://<host>:8081/siebelAIConnectors/mcp/account
```

If multiple MCP Servers are configured, each server is available at its own configured path.

Example:

MCP Server Name	MCP Server Path	MCP Server Endpoint
Account MCP Server	/account	http://<host>:8081/siebelAIConnectors/mcp/account
Product MCP Server	/product	http://<host>:8081/siebelAIConnectors/mcp/product
Service Request MCP Server	/service-request	http://<host>:8081/siebelAIConnectors/mcp/service-request

Use an MCP client, such as Postman, MCP Inspector or another supported MCP client, to connect to the required MCP Server endpoint and verify that the configured tools are available.

If the MCP Server endpoint is not available, verify the following:

- The Siebel AI Connectors runtime started successfully.
- The runtime startup logs show the MCP Server base URL.
- The **server.port** value in **application.yaml** is correct.
- The **mcp-server-path** value in **MCP_CONFIG.json** is correct.

- Each `mcp-server-path` value is unique.
- The MCP Server path in the client request matches the value configured in `MCP_CONFIG.json`.
- Network access to the runtime host and port is available.
- Authentication headers are included if JWT authentication is enabled.

Updating an Existing On-Premises Deployment

The runtime does not dynamically reload configuration changes.

- If `FINAL_SPECIFICATION.json` is updated, review and update the tool mappings in `MCP_CONFIG.json`, and then restart the runtime.
- If `MCP_CONFIG.json` is updated, restart the runtime.
- If `application.yaml` or `OI_CONFIG.json` is updated, restart the runtime.

Deploying Siebel AI Connectors using Siebel Cloud Manager

This section describes the Siebel Cloud Manager payloads used to deploy Siebel AI Connectors in a Kubernetes-based environment.

Siebel AI Connectors expose selected Siebel Open Integration REST APIs as MCP-compatible tools. In an SCM deployment, the runtime configuration artifacts are maintained in a customer Git repository and deployed through Siebel Cloud Manager.

The deployment process includes the following high-level steps:

1. Retrieve the Siebel AI Connectors configuration templates.
2. Register the customer Git repository.
3. Create the required infrastructure.
4. Verify the infrastructure.
5. Deploy Siebel AI Connectors.
6. Check the deployment status.
7. Delete the deployment, if required.
8. Perform incremental updates after configuration changes.

1. Retrieve Configuration Templates

Use this API to retrieve the base configuration templates required for Siebel AI Connectors.

```
API
GET /scm/api/v1.0/siebelAIConnectors/templates HTTP/1.1
Host: https://<scm_host>:<scm_port>
Authorization: Basic <base64_admin_credentials>
```

The response contains a ZIP file with the following template artifacts:

```
Expected Template Artifacts:
application.yaml, MCP_CONFIG.json, OI_CONFIG.json, FINAL_SPECIFICATION.json, logging.properties
```

These files are updated by the customer and committed to the Git repository used by the Siebel AI Connectors deployment.

2. Register Customer Git Repository

Use this API to register the Git repository that contains the Siebel AI Connectors configuration artifacts.

SCM validates access to the repository and returns a `git_id`. This `git_id` is used later in the Siebel AI Connectors deployment payload.

```
API
POST /scm/api/v1.0/git_repositories HTTP/1.1
Host: https://<scm_host>:<scm_port>
Authorization: Basic <base64_admin_credentials>
Content-Type: application/json

API
POST /scm/api/v1.0/git_repositories HTTP/1.1
Host: https://<scm_host>:<scm_port>
Authorization: Basic <base64_admin_credentials>
Content-Type: application/json

Sample HTTPS Git Payload with Self-Signed CA
{
 "git": {
 "git_protocol_type": "https",
 "git_url": "https://<git_host>/<group>/<repo>.git",
 "git_user": "<git_user>",
 "git_accesstoken": "<git_access_token>",
 "git_selfsigned_cacert": "/home/opc/<path_to_ca_certificate>"
 }
}

Sample SSH Git Payload
{
 "git": {
 "git_protocol_type": "ssh",
 "git_url": "git@<git_host>:<group>/<repo>.git",
 "git_ssh_private_key": "/home/opc/.ssh/<private_key_file>"
 }
}

Example Response
{
 "data": {
 "git_id": "<git_id>",
 "git_url": "https://<git_host>/<group>/<repo>.git",
 "git_protocol_type": "https"
 },
 "message": "Git repository registered successfully.",
 "status": "success"
}
```

3. Create Infrastructure

Use this API to create the infrastructure required for the Siebel AI Connectors deployment.

The payload contains the following top-level sections:

Section	Description
operators	Defines operator installation requirements.
kubernetes	Defines the Kubernetes environment where the deployment will run.
git	Defines the GitOps and Helm chart repositories used by SCM.
registry	Defines the container registry used for images.
`security_context`	Optional. Defines custom non-root user and group IDs.

```
API
POST /scm/api/v1.0/infrastructure HTTP/1.1
```

```
Host: https://<scm_host>:<scm_port>
Authorization: Basic <base64_admin_credentials>
Content-Type: application/json

Sample Infrastructure Payload without Security Context
{
 "operators": [
 {
 "coherence_operator":< false | true >
 }
 ],
 "kubernetes": {
 "namespace": "<infra_namespace>",
 "kubernetes_type": "BYO_OCNE",
 "byo_ocne": {
 "kubeconfig_path": "/home/opc/siebel/<kubeconfig_file>.yaml"
 }
 },
 "git": {
 "git_type": "byo_git",
 "byo_git": {
 "git_protocol_type": "https",
 "git_accesstoken": "<git_access_token>",
 "git_user": "<git_user>",
 "git_scm_repo_url": "https://<git_host>/<group>/<siebel_ai_connectors_gitops_repo>",
 "git_scm_repo_branch": "main",
 "git_scm_flux_folder": "flux",
 "git_helm_repo_url": "https://<git_host>/<group>/<helm_chart_repo>",
 "git_helm_repo_branch": "main",
 "skip_flux_bootstrap": false
 }
 },
 "registry": {
 "registry_url": "<registry_host>",
 "registry_user": "<registry_user>",
 "registry_password": "<registry_password>",
 "registry_prefix": "<registry_prefix>"
 }
}
```

Use the following payload when infrastructure and operator setup must use custom non-root IDs:

```
Sample Infrastructure Payload with Security Context
{
 "security_context": {
 "run_as_user": <uid>,
 "run_as_group": <gid>,
 "fs_group": <fs_group_id>
 },
 "operators": [
 {
 "coherence_operator": < false | true >
 }
 ],
 "kubernetes": {
 "namespace": "<infra_namespace>",
 "kubernetes_type": "BYO_OCNE",
 "byo_ocne": {
 "kubeconfig_path": "/home/opc/siebel/<kubeconfig_file>.yaml"
 }
 },
 "git": {
 "git_type": "byo_git",
 "byo_git": {
 "git_protocol_type": "https",
 "git_accesstoken": "<git_access_token>",
```

```
 "git_user": "<git_user>",
 "git_scm_repo_url": "https://<git_host>/<group>/<siebel_ai_connectors_gitops_repo>",
 "git_scm_repo_branch": "main",
 "git_scm_flux_folder": "flux",
 "git_helm_repo_url": "https://<git_host>/<group>/<helm_chart_repo>",
 "git_helm_repo_branch": "main",
 "skip_flux_bootstrap": false
 }
 },
 "registry": {
 "registry_url": "<registry_host>",
 "registry_user": "<registry_user>",
 "registry_password": "<registry_password>",
 "registry_prefix": "<registry_prefix>"
 }
}
```

Sample Response

```
{
 "data": {
 "infra_details": {
 "components_info": {},
 "infra_id": "<infra_id>",
 "infra_status": "creation-in-progress",
 "payload": {
 "git": {
 "byo_git": {
 "git_accesstoken": "********",
 "git_helm_repo_branch": "main",
 "git_helm_repo_url": "https://<git_host>/<group>/<helm_chart_repo>",
 "git_protocol_type": "http",
 "git_scm_flux_folder": "flux",
 "git_scm_repo_branch": "main",
 "git_scm_repo_url": "https://<git_host>/<group>/<siebel_ai_connectors_gitops_repo>",
 "git_user": "<git_user>",
 "skip_flux_bootstrap": false
 },
 "git_type": "byo_git"
 },
 "kubernetes": {
 "byo_ocne": {
 "kubeconfig_path": "/home/opc/siebel/kubeconfig.yaml"
 },
 "kubernetes_type": "BYO_OCNE",
 "namespace": "<infra_namespace>"
 },
 "operators": [
 {
 "coherence_operator": true
 }
 ],
 "registry": {
 "registry_password": "************",
 "registry_prefix": "<registry_prefix>",
 "registry_url": "<registry_host>",
 "registry_user": "<registry_user>"
 }
 },
 "resource_status": {
 "git": "not-validated",
 "kubernetes": "not-validated",
 "operators": [
 {
 "coherence_operator": "not-validated"
 }
```

```
 ],
 "registry": "not-validated"
 },
 "self_link": "https://<scm_host>:<scm_port>/scm/api/v1.0/infrastructure/<infra_id>",
 "stages_info": [
 {
 "end_time": "",
 "log_api_link": "https://<scm_host>:<scm_port>/scm/api/v1.0/infrastructure/<infra_id>/logs/
setup_validation_helmchart",
 "log_location": "",
 "name": "Setup Helm Chart for validation",
 "previous_status": "",
 "stage_name": "setup_validation_helmchart",
 "start_time": "<start_time>",
 "status": ""
 },
 {
 "end_time": "",
 "log_api_link": "https://<scm_host>:<scm_port>/scm/api/v1.0/infrastructure/<infra_id>/logs/
check_job_status_helmchart",
 "log_location": "",
 "name": "Check Job status",
 "previous_status": "",
 "stage_name": "check_job_status_helmchart",
 "start_time": "<start_time>",
 "status": ""
 },
 {
 "end_time": "",
 "log_api_link": "https://<scm_host>:<scm_port>/scm/api/v1.0/infrastructure/<infra_id>/logs/
install_operator_workflow",
 "log_location": "",
 "name": "Install operator workflow",
 "previous_status": "",
 "stage_name": "install_operator_workflow",
 "start_time": "<start_time>",
 "status": ""
 },
 {
 "end_time": "",
 "log_api_link": "https://<scm_host>:<scm_port>/scm/api/v1.0/infrastructure/<infra_id>/logs/
verify_operator_workflow",
 "log_location": "",
 "name": "Verify Operator Installation",
 "previous_status": "",
 "stage_name": "verify_operator_workflow",
 "start_time": "<start_time>",
 "status": ""
 },
 {
 "end_time": "",
 "log_api_link": "https://<scm_host>:<scm_port>/scm/api/v1.0/infrastructure/<infra_id>/logs/
collect_infra_artifacts",
 "log_location": "",
 "name": "Collect Infra artifacts",
 "previous_status": "",
 "stage_name": "collect_infra_artifacts",
 "start_time": "<start_time>",
 "status": ""
 }
 ]
 }
 },
 "message": "Infrastructure successfully created.",
 "status": "success"
```

```
}
```

After this response is returned, use the infra_id value from data.infra_details.infra_id in the subsequent infrastructure verification and Siebel AI Connectors deployment steps.

4. Verify Infrastructure

Use this API to verify that the infrastructure has been created successfully.

```
API
GET /scm/api/v1.0/infrastructure/<infra_id> HTTP/1.1
Host: https://<scm_host>:<scm_port>
Authorization: Basic <base64_admin_credentials>

Sample Response
{
 "data": {
 "infra_details": {
 "components_info": {
 "coherence_operator": true,
 "git": {
 "byo_git": {
 "git_accesstoken": "*************",
 "git_helm_repo_branch": "<helm_repo_branch>",
 "git_helm_repo_url": "https://<git_host>/<group>/<helm_chart_repo>",
 "git_protocol_type": "<http|https|ssh>",
 "git_scm_flux_folder": "<flux_folder>",
 "git_scm_repo_branch": "<gitops_repo_branch>",
 "git_scm_repo_url": "https://<git_host>/<group>/<gitops_repo>",
 "git_user": "<git_user>",
 "skip_flux_bootstrap": false
 },
 "git_type": "byo_git"
 },
 "kubernetes": {
 "byo_ocne": {
 "kubeconfig_path": "/home/opc/siebel/<kubeconfig_file>.yaml"
 },
 "k8s_profile": "/home/opc/siebel/infrastructures/<infra_id>/k8sprofile",
 "kubeconfig_path": "/home/opc/siebel/infrastructures/<infra_id>/kubeconfig.yaml",
 "kubernetes_type": "BYO_OCNE",
 "namespace": "<infra_namespace>"
 },
 "registry": {
 "registry_password": "************",
 "registry_prefix": "<registry_prefix>",
 "registry_url": "<registry_host>",
 "registry_user": "<registry_user>"
 }
 },
 "infra_id": "<infra_id>",
 "infra_status": "completed",
 "payload": {
 "git": {
 "byo_git": {
 "git_accesstoken": "**********",
 "git_helm_repo_branch": "<helm_repo_branch>",
 "git_helm_repo_url": "https://<git_host>/<group>/<helm_chart_repo>",
 "git_protocol_type": "<http|https|ssh>",
 "git_scm_flux_folder": "<flux_folder>",
 "git_scm_repo_branch": "<gitops_repo_branch>",
 "git_scm_repo_url": "https://<git_host>/<group>/<gitops_repo>",
 "git_user": "<git_user>",
 "skip_flux_bootstrap": false
 },
 "git_type": "byo_git"
```

```
 },
 "kubernetes": {
 "byo_ocne": {
 "kubeconfig_path": "/home/opc/siebel/<kubeconfig_file>.yaml"
 },
 "k8s_profile": "/home/opc/siebel/infrastructures/<infra_id>/k8sprofile",
 "kubeconfig_path": "/home/opc/siebel/infrastructures/<infra_id>/kubeconfig.yaml",
 "kubernetes_type": "BYO_OCNE",
 "namespace": "<infra_namespace>"
 },
 "operators": [
 {
 "coherence_operator": true
 }
 ],
 "registry": {
 "registry_password": "*********",
 "registry_prefix": "<registry_prefix>",
 "registry_url": "<registry_host>",
 "registry_user": "<registry_user>"
 }
 },
 "resource_status": {
 "git": "validated",
 "kubernetes": "validated",
 "operators": [
 {
 "coherence_operator": "validated"
 }
 ],
 "registry": "validated"
 },
 "self_link": "https://<scm_host>:<scm_port>/scm/api/v1.0/infrastructure/<infra_id>",
 "stages_info": [
 {
 "end_time": "<end_time>",
 "log_api_link": "https://<scm_host>:<scm_port>/scm/api/v1.0/infrastructure/<infra_id>/logs/
setup_validation_helmchart",
 "log_location": "/home/opc/siebel/infrastructures/<infra_id>/logs/install_connection_chart.log",
 "name": "Setup Helm Chart for validation",
 "previous_status": "",
 "stage_name": "setup_validation_helmchart",
 "start_time": "<start_time>",
 "status": "passed"
 },
 {
 "end_time": "<end_time>",
 "log_api_link": "https://<scm_host>:<scm_port>/scm/api/v1.0/infrastructure/<infra_id>/logs/
check_job_status_helmchart",
 "log_location": "/home/opc/siebel/infrastructures/<infra_id>/logs/validate_connection_chart.log",
 "name": "Check Job status",
 "previous_status": "",
 "stage_name": "check_job_status_helmchart",
 "start_time": "<start_time>",
 "status": "passed"
 },
 {
 "end_time": "<end_time>",
 "log_api_link": "https://<scm_host>:<scm_port>/scm/api/v1.0/infrastructure/<infra_id>/logs/
install_operator_workflow",
 "log_location": "/home/opc/siebel/infrastructures/<infra_id>/logs/operators.log",
 "name": "Install operator workflow",
 "previous_status": "",
 "stage_name": "install_operator_workflow",
 "start_time": "<start_time>",
 "status": "passed"
```

```
 },
 {
 "end_time": "<end_time>",
 "log_api_link": "https://<scm_host>:<scm_port>/scm/api/v1.0/infrastructure/<infra_id>/logs/
verify_operator_workflow",
 "log_location": "/home/opc/siebel/infrastructures/<infra_id>/logs/operators.log",
 "name": "Verify Operator Installation",
 "previous_status": "",
 "stage_name": "verify_operator_workflow",
 "start_time": "<start_time>",
 "status": "passed"
 },
 {
 "end_time": "<end_time>",
 "log_api_link": "https://<scm_host>:<scm_port>/scm/api/v1.0/infrastructure/<infra_id>/logs/
collect_infra_artifacts",
 "log_location": "/home/opc/siebel/infrastructures/<infra_id>/logs/artifacts.log",
 "name": "Collect Infra artifacts",
 "previous_status": "",
 "stage_name": "collect_infra_artifacts",
 "start_time": "<start_time>",
 "status": "passed"
 }
 ]
 }
 },
 "message": "Infrastructure details retrieved successfully.",
 "status": "success"
}
```

After verification completes successfully, the `infra_status` value is `completed`, and the `resource_status` values for Git, Kubernetes, operators, and registry are marked as `validated`. Use the `infra_id` value in the Siebel AI Connectors deployment payload.

5. Deploy Siebel AI Connectors

Use this API to deploy Siebel AI Connectors using the prepared infrastructure and registered Git repository. The deployment payload references the Git repository that contains the runtime configuration files.

```
API
POST /scm/api/v1.0/siebelAIConnectors HTTP/1.1
Host: https://<scm_host>:<scm_port>
Authorization: Basic <base64_admin_credentials>
Content-Type: application/json

Deployment Payload
{
 "name": "<mcp_deployment_namespace>",
 "infra_id": "<infra_id>",
 "byo_ns": false,
 "security_context": {
 "run_as_user": <uid>,
 "run_as_group": <gid>,
 "fs_group": <fs_group_id>
 },
 "siebel_mcp": {
 "git": {
 "git_id": "<git_id>",
 "git_repo_branch": "main",
 "git_config_folder": "siebel-mcp"
 },
 "java_truststore": {
 "cacerts_path": "/home/opc/siebel/cacerts",
 "password": "<truststore_password>"
 }
```

```
 }
}
```

Optionally, If `byo_ns` is set to "`true`", ensure that the Siebel AI Connector namespace is created before executing this step. Then, specify the created namespace in the Siebel AI Connector deployment payload by setting the "`name`" field to the namespace value.

Example Response


```
{
 "data": {
 "ansible_chart_name": "ansible-<deployment_name>-<deploy_id_lowercase>",
 "app_dir": "/home/opc/siebel/microservices/siebel-ai-connectors/<deploy_id>",
 "byo_ns": false,
 "deploy_id": "<deploy_id>",
 "deploy_status": "creation-in-progress",
 "infra_id": "<infra_id>",
 "name": "<mcp_deployment_namespace>",
 "namespace": "<mcp_deployment_namespace>",
 "product_name": "siebel-ai-connectors",
 "security_context": {
 "fs_group": <fs_group_id>,
 "run_as_group": <gid>,
 "run_as_user": <uid>
 },
 "self_link": "https://<scm_host>:<scm_port>/scm/api/v1.0/siebelAIConnectors/<deploy_id>",
 "siebel_mcp": {
 "git": {
 "git_config_folder": "siebel-mcp",
 "git_id": "<git_id>",
 "git_json": {
 "branch": "main",
 "config_folder": "siebel-mcp",
 "protocol_type": "https",
 "url": "https://<git_host>/<group>/<repo>",
 "user": "<git_user>"
 },
 "git_protocol_type": "https",
 "git_repo_branch": "main",
 "git_selfsigned_cacert": "",
 "git_siebel_mcp_directory": "siebel-mcp",
 "git_ssh_private_key": "",
 "git_url": "https://<git_host>/<group>/<repo>",
 "git_user": "<git_user>"
 },
 "java_truststore": {
 "cacerts_path": "/home/opc/siebel/cacerts",
 "helm_values": {
 "checksum": "<truststore_checksum>",
 "enabled": true,
 "fileName": "cacerts",
 "mountPath": "/opt/mcp/truststore",
 "password": "",
 "passwordSecretKey": "storepass",
 "passwordSecretName": "",
 "secretKey": "cacerts",
 "secretName": ""
 },
 "password": "<masked_or_configured_truststore_password>"
 }
 },
 "stages": [
 {
 "name": "Pre Deploy",
 "stage_name": "pre_deploy",
 "stage_description": "Create pre requisites and other preparations",
```

```
 "status": ""
 },
 {
 "name": "Prepare GitOps",
 "stage_name": "prepare_gitops",
 "stage_description": "Generate Flux definitions, Prepare helmchart to git and Prepare GitOps",
 "status": ""
 },
 {
 "name": "Verify Deployment",
 "stage_name": "verify_deployment",
 "stage_description": "Verify HR Deployment",
 "status": ""
 },
 {
 "name": "Publish Urls",
 "stage_name": "publish_urls",
 "stage_description": "Publish Application Urls",
 "status": ""
 }
 ]
 },
 "message": "ansible playbook for siebel-ai-connectors ansible playbook for executed successfully",
 "status": "success"
}
```

6. Get Siebel AI Connectors Deployment Status

Use this API to retrieve the current status of a Siebel AI Connectors deployment.

```
API
GET /scm/api/v1.0/siebelAIConnectors/<deploy_id> HTTP/1.1
Host: https://<scm_host>:<scm_port>
Authorization: Basic <base64_admin_credentials>

Example Response
{
 "data": {
 "ansible_chart_name": "ansible-<deployment_name>-<deploy_id_lowercase>",
 "app_dir": "/home/opc/siebel/microservices/siebel-ai-connectors/<deploy_id>",
 "byo_ns": false,
 "deploy_id": "<deploy_id>",
 "deploy_status": "<creation-in-progress|completed|failed|Rerun-In-Progress>",
 "infra_id": "<infra_id>",
 "name": "<mcp_deployment_namespace>",
 "namespace": "<mcp_deployment_namespace>",
 "product_name": "siebel-ai-connectors",
 "security_context": {
 "fs_group": <fs_group_id>,
 "run_as_group": <gid>,
 "run_as_user": <uid>
 },
 "self_link": "https://<scm_host>:<scm_port>/scm/api/v1.0/siebelAIConnectors/<deploy_id>",
 "siebel_mcp": {
 "git": {
 "git_id": "<git_id>",
 "git_config_folder": "siebel-mcp",
 "git_repo_branch": "main",
 "git_protocol_type": "https",
 "git_url": "https://<git_host>/<group>/<repo>",
 "git_user": "<git_user>"
 },
 "java_truststore": {
 "cacerts_path": "/home/opc/siebel/cacerts",
 "password": "<masked_or_configured_truststore_password>"
 }
```

```
 },
 "stages": [
 {
 "name": "Pre Deploy",
 "stage_name": "pre_deploy",
 "status": "<passed|failed|in-progress>",
 "log_api_link": "https://<scm_host>:<scm_port>/scm/api/v1.0/siebelAIConnectors/<deploy_id>/logs/
pre_deploy",
 "log_location": "/home/opc/siebel/microservices/siebel-ai-connectors/<deploy_id>/logs/pre_deploy.log"
 },
 {
 "name": "Prepare GitOps",
 "stage_name": "prepare_gitops",
 "status": "<passed|failed|in-progress>",
 "log_api_link": "https://<scm_host>:<scm_port>/scm/api/v1.0/siebelAIConnectors/<deploy_id>/logs/
prepare_gitops",
 "log_location": "/home/opc/siebel/microservices/siebel-ai-connectors/<deploy_id>/logs/prepare_gitops.log"
 },
 {
 "name": "Verify Deployment",
 "stage_name": "verify_deployment",
 "status": "<passed|failed|in-progress>",
 "log_api_link": "https://<scm_host>:<scm_port>/scm/api/v1.0/siebelAIConnectors/<deploy_id>/logs/
verify_deployment",
 "log_location": "/home/opc/siebel/microservices/siebel-ai-connectors/<deploy_id>/logs/
verify_deployment.log"
 },
 {
 "name": "Publish Urls",
 "stage_name": "publish_urls",
 "status": "<passed|failed|in-progress>"
 }
 ]
 },
 "message": "success",
 "status": "success"
}
```

7. Delete Siebel AI Connectors Deployment

Use this API to delete an existing Siebel AI Connectors deployment.

```
API
DELETE /scm/api/v1.0/siebelAIConnectors/<deploy_id> HTTP/1.1
Host: https://<scm_host>:<scm_port>
Authorization: Basic <base64_admin_credentials>

Example Response
{
 "data": {
 "archive_dir": "/home/opc/siebel/microservices/siebel_mcp/archive/<deploy_id>",
 "deleted": true,
 "deploy_id": "<deploy_id>"
 },
 "message": "Deletion of deploy_id <deploy_id> of siebel_mcp successfull",
 "status": "success"
}
```

8. Perform Incremental Updates

Perform an incremental update after changes are made to the Siebel AI Connectors configuration files in the customer Git repository.

Configuration files may include either of the following files:

- application.yaml
- MCP_CONFIG.json
- OI_CONFIG.json
- FINAL_SPECIFICATION.json
- logging.properties

Steps:

1. Update the required configuration files in the customer Git repository.
2. Commit and push the customer Git changes.
3. Go to the infrastructure Helm chart repository.

   ```
   cd /home/opc/siebel/infrastructures/<infra_id>/<helmchart_repo>/
   git pull
   ```

4. Change to the image builder chart directory for the deployment.

   ```
   cd <deploy_name>-<deploy_id>/siebel-mcp
   ```

5. Edit `Chart.yaml`.

   ```
   vi Chart.yaml
   ```

6. Increment the chart version.

   ```
   version: 0.1.2
   ```

7. Commit and push the Helm chart change.

   ```
   git add Chart.yaml
   git commit -m "<incremental_update_message>"
   git push
   ```

Expected Result

After the Helm chart version is updated and pushed:

1. Flux reconciliation detects the Helm chart version change.
2. The MCP image builder job restarts.
3. A new Siebel AI Connectors application image is built using the latest committed customer Git content.
4. The updated MCP application starts with the new image.

Important Notes

1. Do not commit credentials, access tokens, registry passwords, truststore passwords, private keys, or other secrets into Git.
2. Ensure that all customer Git changes are committed and pushed before incrementing the Helm chart version.
3. Increment only the chart version required to trigger the image builder update.

Placeholder Reference

Placeholder	Description
<scm_host>	Siebel Cloud Manager host name or IP address.
<scm_port>	Siebel Cloud Manager port.

Placeholder	Description
<infra_id>	Infrastructure ID returned by the infrastructure creation API.
<deploy_id>	Deployment ID returned by the Siebel AI Connectors deployment API.
<git_id>	Git repository ID returned by the Git registration API.
<git_host>	Git server host.
<group>	Git group, project, or organization path.
<repo>	Git repository name.
<git_user>	Git user name.
<git_access_token>	Git access token.
<infra_namespace>	Kubernetes namespace used for infrastructure resources.
<mcp_deployment_namespace>	Namespace or deployment name used for Siebel AI Connectors.
<registry_host>	Container registry host.
<registry_user>	Container registry user.
<registry_password>	Container registry password or authentication token.
<registry_prefix>	Registry namespace or prefix.
<uid>	Runtime user ID.
<gid>	Runtime group ID.
<fs_group_id>	File system group ID.
<truststore_password>	Java truststore password.
<kubeconfig_file>	Kubernetes kubeconfig file available on the SCM host.

Validating Runtime Behavior

Runtime behavior includes startup, configuration loading, MCP Server creation, tool invocation, and validation.

Runtime Startup

When the runtime starts, it performs the following actions:

1. Reads `application.yaml`.
2. Resolves the configured paths for `MCP_CONFIG.json`, `FINAL_SPECIFICATION.json`, `logging.properties`, and `OI_CONFIG.json`.
3. Loads logging configuration.
4. Loads Open Integration server configuration from `OI_CONFIG.json`.
5. Parses `FINAL_SPECIFICATION.json`.
6. Builds MCP tool definitions from the selected Open Integration REST API operations.
7. Loads MCP Server definitions from `MCP_CONFIG.json`.
8. Maps tools to MCP Servers.
9. Starts MCP Server endpoints.

Expected startup indicators include:

```
Successfully parsed OpenAPI spec file:<Path>/FINAL_SPECIFICATION.json
Loaded Open Integration servers: [<server-name-1>, <server-name-2>]; default server: <server-name>
Processing Tool : [<tool-name>] , Method : [<method>] , Path: [<path>]
Using Open Integration server [<server-name>] for tool [<tool-name>]
Adapted [<count>] tools from OpenAPI spec
MCP Servers started with base url: <host>:<port>/siebelAIConnectors/mcp
```

If a server URL in `FINAL_SPECIFICATION.json` differs from the server URL configured in `OI_CONFIG.json`, the runtime logs a warning and uses the configured Open Integration server URL.

Startup Validation

After startup, validate that the runtime loaded all required artifacts successfully.

Validation Check	Expected Result
`application.yaml` is loaded.	Runtime starts with configured host, port, security settings, and file paths.
`logging.properties` is loaded.	Logs use the configured handler, format, and logging levels.
`OI_CONFIG.json` is loaded.	Runtime logs the configured Open Integration servers and default server.
`FINAL_SPECIFICATION.json` is parsed.	Runtime logs successful parsing of the tool specification file.
MCP tools are generated.	Runtime logs the tools processed from the selected operations.
`MCP_CONFIG.json` is parsed.	Runtime creates MCP Servers with configured endpoint paths.
MCP Server endpoints are started.	Runtime logs the MCP base URL.

Tool Discovery and Invocation Validation

After startup validation, validate that MCP tools can be discovered and invoked.

Validation Check	Expected Result
MCP client connects to the MCP Server endpoint.	Connection succeeds.
Configured tools are available for discovery.	Tools listed in **`MCP_CONFIG.json`** are visible through the MCP Server.
Sample MCP tool is invoked.	Runtime identifies the tool and resolves the Open Integration server.
Open Integration REST API is invoked.	Runtime sends the downstream request to the configured Open Integration server.
Response is returned to MCP client.	MCP client receives the response from the tool invocation.

For secured deployments, also validate:

Validation Check	Expected Result
Request without token is sent.	Request is rejected.
Request with invalid token is sent.	Request is rejected.

Validation Check	Expected Result
Request with valid token and required scope is sent.	Request is accepted.
Token contains expected issuer, audience, and scope.	JWT validation succeeds.
Open Integration authentication is aligned.	Downstream Open Integration request succeeds.

Updating Existing Configurations

The runtime does not dynamically reload configuration artifacts. Restart the runtime after updating any of the following files:

- **FINAL_SPECIFICATION.json**
- **MCP_CONFIG.json**
- **application.yaml**
- **OI_CONFIG.json**
- **logging.properties**

It is critical to note that if the **FINAL_SPECIFICATION.json** is updated, then you must also review and if needed update **MCP_CONFIG.json** if tools were added, removed or renamed.

Troubleshooting Validation Issues

Use the following checks when runtime startup or tool invocation fails.

Issue	Check
Runtime does not start.	Verify Java version, runtime classpath, **mcp-runtime-<version>.jar**, **libs/**, and **application.yaml**.
Configuration file not found.	Verify **mcp.server-config-file**, **mcp.final-spec-file**, **mcp.logging-properties-file**, and **mcp.open-int-config-file**.
No tools are available.	Verify **x-tools**, operation-level **x-tool-name**, and **MCP_CONFIG.json** tool mappings.
MCP Server endpoint is missing.	Verify **mcp-server-path** in **MCP_CONFIG.json**.
Tool mapping fails.	Verify every **mcp-server-tools** value exists in **FINAL_SPECIFICATION.json**.
Open Integration server is not resolved.	Verify operation-level **x-server-name**, **OI_CONFIG.json** server names, and **default-server**.
Request body validation fails.	Verify **requestBody**, **components.schemas**, **required**, and **additionalProperties**.
Parameter validation fails.	Verify parameter **in**, **name**, **required**, and **schema**.
JWT validation fails.	Verify **issuer**, **audience**, **required-scope**, **jwks-uri**, and token claims.
Open Integration invocation fails.	Verify Open Integration host, port, context, authentication, certificate trust, and network connectivity.

Configure Open Integration Logging

This topic describes the how you can configure Open Integration logging. It includes the following topics:

- *Open Integration Logging Overview*
- *Enable Package Level Logging*
- *Logging for Generated UI and Data APIs*
- *Apply and Remove Diagnostic Changes*
- *Manage Diagnostic Logs in Production*
- *Troubleshooting Logging Configuration*
- *Example log4j2.xml Configuration*

Open Integration Logging Overview

You can configure where Open Integration writes logs and how much detail it records by updating `log4j2.xml` or setting the logging environment variables.

The `logs` directory can be local or mounted from a remote location. By default, Open Integration writes:

- Application logs to `logs/siebel-openint.log` and the console.
- Dead-letter queue (DLQ) logs to `logs/siebel-openint-dlq.log`.

To write both application and DLQ logs only to the console, you must set `OI_CONSOLE_ONLY=true`. Open Integration does not create log files in this mode.

The following levelsare supported:

Log Level	Description
ERROR	Records error messages only. This is the default.
WARN	Recordswarning and error messages.
INFO	Recordsinformational, warning, and error messages.
DEBUG	Recordsdebug, informational, warning, and error messages.
TRACE	Recordstrace, debug, informational, warning, and error messages.
ALL	Recordsall log messages.

The `OI_LOG_LEVEL` environment variable controls application logging. The `DLQ_LOG_LEVEL` environment variable controls DLQ logging independently. Each log level defaults to `ERROR` when its environment variable is not set. For example:

```
export OI_LOG_LEVEL=INFO
export DLQ_LOG_LEVEL=INFO
export OI_CONSOLE_ONLY=true
```

You must restart Open Integration after you add or change `OI_LOG_LEVEL`, `DLQ_LOG_LEVEL`, or `OI_CONSOLE_ONLY` so that the process reads the updated values. However, you can update `log4j2.xml` without restarting Open Integration. For more information, see Fixed Log Levels and Reload.

> **Note:** For SCM-specific logging configuration and troubleshooting instructions, see the applicable SCM guides.

Routing Log Output

You can control where Open Integration writes application and DLQ logs by setting `OI_CONSOLE_ONLY`.

Setting	Application Logs	DLQ Logs
`OI_CONSOLE_ONLY` is unset or is not true	Written to `logs/siebel-openint.log` and the console, subject to the configured console threshold.	Written to `logs/siebel-op`
`OI_CONSOLE_ONLY=true`	Written only to the console.	Written only to the console.

Application logging and DLQ logging operate independently. Set `OI_LOG_LEVEL` to control application logging and `DLQ_LOG_LEVEL` to control DLQ logging. Changing either variable does not affect the other.

For package-specific logging, the logger level determines which events the logger accepts. Its `AppenderRef` elements determine where those events are written. To display DEBUG or TRACE events in the console, set both the package logger level and its console appender-reference level accordingly. Adding only a package level can result in events being accepted by the logger but filtered before they reach the console.

Adding a package logger does not create a separate log file. When file logging is enabled, package events use FileLogger and appear in `logs/siebel-openint.log` with other application events.

Correlating Log Records for a Request

You can use the request context in each log record to trace a request across Open Integration components.

Pattern Field	Description
`%X{operation}`	The operation being performed.
`%X{channel}`	The channel through which the request was received or processed.
`%X{correlationId}`	The identifier that links records for the same request or flow.
`%X{sourceSystem}`	The calling or source system, when available.
`%X{userAgent}`	The client user-agent information, when available.
`%t`	The thread that generated the record.
`%logger{36}`	The logger name, abbreviated to fit within 36 characters.

To investigate a request, search for its `correlationId`. Then use `operation` and `channel` to follow the request across packages.

> **Note:** Startup records and some background tasks do not contain request context. So, empty context fields do not necessarily indicate an error.

Enable Package Level Logging

You can enable DEBUG or TRACE logging for a specific package by adding a Logger element to log4j2.xml. This lets you collect diagnostic details while keeping the root application logger at INFO or ERROR.

The supplied configuration provides environment variables for the root application logger and DLQ logger only. Configure package-level logging directly in the XML.

A package logger applies to the named package and its subpackages. You must select the narrowest package that covers the issue. Start with DEBUG and use TRACE only when you need more detail.

Select a Logger Package

Use the applicable package name in the Logger element's name attribute.

Area	Logger Package
Integration execution, steps, transformations, workflows, and configuration	`com.siebel.openint.integrations`
Integration step construction	`com.siebel.openint.integrations.steps`
Integration workflow execution	`com.siebel.openint.integrations.workflow`
Integration transformations, including JSONata	`com.siebel.openint.integrations.transformation`
Transformation scripts	`com.siebel.openint.integrations.transformationscript`
Integration configuration loading	`com.siebel.openint.integrations.config`
External REST calls and their OpenID Connect (OIDC) security	`com.siebel.openint.integrations.rest`
OIDC security for external REST calls	`com.siebel.openint.integrations.rest.security`
Generated Open Integration REST endpoints	`com.siebel.openint.resource`
UI sessions, requests, security, and connection infrastructure	`com.siebel.uiinfra`
UI request processing	`com.siebel.uiinfra.request`
UI connection management	`com.siebel.uiinfra.connmgr`
UI session management	`com.siebel.uiinfra.sessmgr`
UI security	`com.siebel.uiinfra.security`
REST connection, metadata, context, and startup infrastructure	`com.siebel.restinfra`
REST connection management	`com.siebel.restinfra.connmgr`
REST metadata processing	`com.siebel.restinfra.metadata`

Area	Logger Package
Domain-specific language (DSL) flow execution	`com.siebel.openint.dsl`
Siebel flow DSL execution	`com.siebel.openint.dsl.siebelflow`
Proxy request handling and clients	`com.siebel.openint.proxy`
Open Integration channels	`com.siebel.openint.channels`
Open Integration handlers	`com.siebel.openint.handlers`
Siebel data access	`com.siebel.data`
Siebel integration utilities and outbound REST support	`com.siebel.integration`

For example:

If the issue is with	Add this logger at DEBUG first
An integration, mapping, step, retry, or workflow	`com.siebel.openint.integrations`
An outbound call to another REST service	`com.siebel.openint.integrations.rest`
A UI session, login, connection, or UI request	`com.siebel.uiinfra`
REST connection setup or API metadata	`com.siebel.restinfra`
A generated Open Integration API request	`com.siebel.openint.resource`
A data-access operation	`com.siebel.data`

Use TRACE only if DEBUG does not provide enough detail to diagnose the issue.

Prerequisites

Before you edit the installed configuration:

- Save a copy of the current `log4j2.xml` file so that you can restore it.
- Confirm whether the deployment uses file logging or `OI_CONSOLE_ONLY=true`.
- Select the package to diagnose and define a short diagnostic period.
- Identify every instance that can receive the affected request. Apply the configuration to those instances or route the test request to the configured instance.

Configure a Package Logger for File Logging

Use this procedure when `OI_CONSOLE_ONLY` is not set or is not true.

1. Open `<install-location>/openint-deploy/target/log4j2.xml`.
2. Locate `DefaultArbiter` under `Loggers`.

3. Add the package logger immediately before the `Root` element. For example, add the following logger to enable integration diagnostics:

```
<Logger name="com.siebel.openint.integrations"
 level="DEBUG"
 additivity="false">
 <AppenderRef ref="FileLogger"/>
 <AppenderRef ref="Console" level="DEBUG"/>
</Logger>
```

   FileLogger writes the records to `logs/siebel-openint.log`. The Console reference also sends the selected diagnostic records to the console. Setting `additivity="false"` prevents the root logger from writing the same event again.
4. Save the file and follow the steps in *Apply and Remove Diagnostic Changes*.

   You can keep the rest of the application at INFO by setting the following environment variable before starting Open Integration:

```
export OI_LOG_LEVEL=INFO
```

5. Restart Open Integration if you change this environment variable while the process is running.

   To collect `TRACE` records only for outbound REST calls, use this package logger instead:

```
<Logger name="com.siebel.openint.integrations.rest"
 level="TRACE"
 additivity="false">
 <AppenderRef ref="FileLogger"/>
 <AppenderRef ref="Console" level="TRACE"/>
</Logger>
```

Configure a Package Logger for Console-Only Logging

Use this procedure when OI_CONSOLE_ONLY=true.

1. Open `<install-location>/openint-deploy/target/log4j2.xml`.
2. Locate the `EnvironmentArbiter` branch with `propertyName="OI_CONSOLE_ONLY"` and `propertyValue="true"`.
3. Add the package logger immediately before the `Root` element. Use only the Console appender:

```
<EnvironmentArbiter propertyName="OI_CONSOLE_ONLY" propertyValue="true">
 <Logger name="DLQLogger"
 level="${env:DLQ_LOG_LEVEL:-ERROR}"
 additivity="false">
 <AppenderRef ref="Console" level="${env:DLQ_LOG_LEVEL:-ERROR}"/>
 </Logger>
 <Logger name="com.siebel.openint.integrations"
 level="DEBUG"
 additivity="false">
 <AppenderRef ref="Console" level="DEBUG"/>
 </Logger>

 <Root level="${env:OI_LOG_LEVEL:-ERROR}" additivity="false">
 <AppenderRef ref="Console" level="${env:OI_LOG_LEVEL:-ERROR}"/>
 </Root>
</EnvironmentArbiter>
```

4. Save the file and follow the steps in *Apply and Remove Diagnostic Changes*.

> **Note:** FileLogger does not exist in console-only mode. Do not reference it in the `EnvironmentArbiter` branch. If the deployment can use either logging mode, add the corresponding package logger to both branches with the appropriate appenders.

Configure Multiple Package Loggers

You can configure multiple package loggers. The most specific matching package controls an event.

For example, the following configuration enables DEBUG for integrations and TRACE for outbound REST calls:

```
<Logger name="com.siebel.openint.integrations"
 level="DEBUG"
 additivity="false">
 <AppenderRef ref="FileLogger"/>
 <AppenderRef ref="Console" level="DEBUG"/>
</Logger>

<Logger name="com.siebel.openint.integrations.rest"
 level="TRACE"
 additivity="false">
 <AppenderRef ref="FileLogger"/>
 <AppenderRef ref="Console" level="TRACE"/>
</Logger>
```

Use the appropriate appender configuration for every logger in the branch. Retain `additivity="false"` to prevent duplicate records. In file mode, include FileLogger and include Console when you also need console diagnostics. Set the console `appender-reference` threshold to allow the selected log level.

Logging for Generated UI and Data APIs

Generated UI REST APIs and Data APIs share the `com.siebel.openint.resource` package. Add this logger to diagnose either API group. It enables logging for both groups.

To investigate the supporting infrastructure, use:

- `com.siebel.uiinfra` for UI sessions, requests, security, and connection management.
- `com.siebel.restinfra` for REST connections, metadata, context, and startup infrastructure.

You cannot configure separate package-level logging for generated UI APIs and generated Data APIs in the current package structure.

> **Note:** Do not use `OpenintegrationV10Ui` or `OpenintegrationV10Data` as logger package names. These are naming prefixes, not Java packages.

Apply and Remove Diagnostic Changes

You can set a fixed level, such as `level="DEBUG"`, on a Logger or Root element in `log4j2.xml`. Use `OI_LOG_LEVEL` and `DLQ_LOG_LEVEL` for temporary root or DLQ changes, and package loggers for targeted diagnostics.

With `monitorInterval="300"`, `Log4j2` detects configuration-file changes within approximately five minutes.

To apply and verify a diagnostic change:

1. Edit `<install-location>/openint-deploy/target/log4j2.xml`.

2. Validate the XML syntax. If `xmllint` is available, run:
   ```
   xmllint --noout log4j2.xml
   ```

   This command checks XML syntax. It does not verify whether the package applies to the request you are investigating.
3. Wait up to five minutes for the configuration to reload. If you need the change immediately, restart Open Integration.
4. Reproduce the affected request or integration execution.
5. Review `logs/siebel-openint.log` for file logging, or the service or container console for console-only logging.
6. Remove the temporary package logger or change its level to INFO or ERROR after diagnosis. Do not leave TRACE or ALL enabled in a production deployment.

Note: Keep temporary diagnostic changes outside the checked-in source configuration. Changes under `<install-location>/openint-deploy/target` can be replaced during packaging or deployment. If a logger must remain after redeployment, apply it through the deployment's managed configuration process.

Manage Diagnostic Logs in Production

Use the narrowest package and the shortest practical diagnostic period. DEBUG, TRACE, and ALL can substantially increase log volume, disk usage, console ingestion costs, and search time. Do not leave TRACE or ALL enabled in production after diagnosis.

Diagnostic records can contain request metadata, identifiers, payload fragments, endpoint details, or other customer data. Restrict access and review logs for sensitive information before sharing them outside your organization. Do not include credentials, passwords, access tokens, private keys, or unredacted secrets in log messages or diagnostic notes.

Each rolling file rolls over at 100 MB. The supplied `DefaultRolloverStrategy` uses `fileIndex="nomax"`, which does not limit the number of retained rollover files. Monitor available storage and manage retention according to your operational policy.

In deployments with multiple instances, each instance writes to its own local path unless shared or aggregated logging is configured. Collect records from every instance that handled the request.

Troubleshooting Logging Configuration

The following table lists common logging issues and the actions you can take to resolve them:

Issue	Solution
No DEBUG records appear.	Verify the package name and confirm that the application executed the relevant code. Check that the logger level and appender-reference threshold allow DEBUG.
Records appear in the file but not in the console.	Verify that the package logger includes a Console reference with a level that allows the diagnostic records. Changing a package level does not change the root console threshold.

Issue	Solution
FileLogger cannot be found.	Check whether the logger was added to the console-only EnvironmentArbiter branch. Use only Console in that branch.
Records are duplicated.	Verify that the package logger uses additivity="false" and doesnot forward the same event through the root logger.
The change has no effect after five minutes.	Verify that the process uses the edited file. Check the console for Log4j2 configuration warnings, validate the XML, and restart Open Integration if necessary.
Logging changes affect only some requests.	Verify that every instance handling those requests has the required configuration.
The change disappears after redeployment.	Apply the change through the deployment-managed source or mounted configuration. Files under the deployment target can be replaced during packaging or deployment.

Example log4j2.xml Configuration

```
<Configuration status="WARN" name="Logging" monitorInterval="300">
 <Appenders>
 <Console name="Console" target="SYSTEM_OUT">
 <PatternLayout>
 <pattern>
 %d{yyyy-MM-dd HH:mm:ss.SSS} | %-5level | %X{operation} | %X{channel}:%X{correlationId} | %X{sourceSystem}:
%X{userAgent} | %t | %logger{36} - %msg%n
 </pattern>
 </PatternLayout>
 </Console>
 <Select>
 <EnvironmentArbiter propertyName="OI_CONSOLE_ONLY" propertyValue="true"/>
 <DefaultArbiter>
 <RollingFile name="FileLogger"
 fileName="logs/siebel-openint.log"
 filePattern="logs/siebel-openint-%d{yyyy-MM-dd}-%i.log">
 <PatternLayout>
 <pattern>
 %d{yyyy-MM-dd HH:mm:ss.SSS} | %-5level | %X{operation} | %X{channel}:%X{correlationId} | %X{sourceSystem}:
%X{userAgent} | %t | %logger{36} - %msg%n
 </pattern>
 </PatternLayout>
 <Policies>
 <SizeBasedTriggeringPolicy size="100 MB"/>
 </Policies>
 <DefaultRolloverStrategy fileIndex="nomax" />
 </RollingFile>
 <RollingFile name="DLQFileLogger"
 fileName="logs/siebel-openint-dlq.log"
 filePattern="logs/siebel-openint-dlq-%d{yyyy-MM-dd}-%i.log">
 <PatternLayout>
 <pattern>
```

```
 %d{yyyy-MM-dd HH:mm:ss.SSS} | %-5level | %X{operation} | %X{channel}:%X{correlationId} | %X{sourceSystem}:
%X{userAgent} | %t | %logger{36} - %msg%n
 </pattern>
 </PatternLayout>
 <Policies>
 <SizeBasedTriggeringPolicy size="100 MB"/>
 </Policies>
 <DefaultRolloverStrategy fileIndex="nomax" />
 </RollingFile>
 </DefaultArbiter>
 </Select>
 </Appenders>
 <Loggers>
 <Select>
 <EnvironmentArbiter propertyName="OI_CONSOLE_ONLY" propertyValue="true">
 <Logger name="DLQLogger" level="${env:DLQ_LOG_LEVEL:-ERROR}" additivity="false">
 <AppenderRef ref="Console" level="${env:DLQ_LOG_LEVEL:-ERROR}"/>
 </Logger>
 <Root level="${env:OI_LOG_LEVEL:-ERROR}" additivity="false">
 <AppenderRef ref="Console" level="${env:OI_LOG_LEVEL:-ERROR}"/>
 </Root>
 </EnvironmentArbiter>
 <DefaultArbiter>
 <Logger name="DLQLogger" level="${env:DLQ_LOG_LEVEL:-ERROR}" additivity="false">
 <AppenderRef ref="DLQFileLogger"/>
 </Logger>
 <Root level="${env:OI_LOG_LEVEL:-ERROR}" additivity="false">
 <AppenderRef ref="FileLogger"/>
 <AppenderRef ref="Console" level="${env:OI_LOG_LEVEL:-ERROR}"/>
 </Root>
 </DefaultArbiter>
 </Select>
 </Loggers>
</Configuration>
```