Siebel

Anywhere Administration Guide

January 2020

Siebel
Anywhere Administration Guide

January 2020

Part Number: F87440-01

Copyright © 1994, 2020, Oracle and/or its affiliates.

Authors: Siebel Information Development Team

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish, or display in any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

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

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

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

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

Oracle and Java are registered trademarks of Oracle Corporation and/or its affiliates. Other names may be trademarks of their respective owners.

Intel and Intel Xeon are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. AMD, Opteron, the AMD logo, and the AMD Opteron logo are trademarks or registered trademarks of Advanced Micro Devices. UNIX is a registered trademark of The Open Group.

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

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

Contents

Preface i

1 What's New in This Release 1

What's New in Siebel Anywhere Administration Guide, Siebel CRM 20.1 Update 1
What's New in Siebel Anywhere Administration Guide, Siebel CRM 19.1 Update 1

2 Overview of Siebel Anywhere 3

Overview of Siebel Anywhere 3
Features and Benefits of Siebel Anywhere 3
Roadmap for Using Siebel Anywhere 4
About the Siebel Anywhere User Interface Features 5
About Siebel Anywhere File Attachments 7
About Siebel Anywhere Version Settings 7
Siebel Anywhere Terminology 12
Siebel Anywhere Deployment Recommendations 18

3 Planning and Preparing to Use Siebel Anywhere 19

Planning and Preparing to Use Siebel Anywhere 19
Process of Planning and Preparing to Create Upgrade Kits 19
Determining Your Upgrade Requirements 20
Modifying and Creating Siebel Anywhere Configurations 32
Setting Up Custom Siebel Anywhere Upgrade Components 36
Verifying Client Connections to the Siebel File System 40
Verifying the Availability of Siebel Anywhere 40
Preparing Upgrade Kit Contents 40
Creating a Delta Patch File 45
Synchronizing Server Components 48

4 Defining Upgrade Kits 49

Defining Upgrade Kits 49
About Defining Upgrade Kits 49

Running the Upgrade Kit Wizard 50
Defining a Siebel Upgrade Wizard Upgrade Kit 51
Defining a Siebel Configuration File (CFG) Upgrade Kit 52
Defining a Siebel Database Schema Upgrade Kit 55
Defining a Siebel Client Executables Upgrade Kit 58
Roadmap for Distributing a Siebel Innovation Pack or Patchset Release 62
Roadmap for Creating a Delta Installation Siebel Client Executables Upgrade Kit 63
Defining a Siebel Patchset Upgrade Kit 64
Defining a Siebel Repository Export Import Upgrade Kit 67
Defining a Third Party Software Upgrade Kit 70
Defining a Siebel Workspace Seed Import 73
Defining a Siebel Client Customer Revisions Upgrade Kit 77
Defining a Custom Component Upgrade Kit 82
Viewing Upgrade Kit Properties 83

5 Retrieving, Installing, and Testing Upgrade Kits 89

Retrieving, Installing, and Testing Upgrade Kits 89
Process of Retrieving, Installing, and Testing Upgrade Kits 89
Retrieving and Installing Upgrade Kits 90
Starting the Upgrade Wizard 94
Retrieving Upgrade Kits for Siebel Regional Node Servers 95
Testing Upgrade Kits 97
Uninstalling a Delta Patch 98

6 Activating, Applying, and Distributing Upgrade 101

Activating, Applying, and Distributing Upgrade Kits 101
Process of Completing Upgrade Kit Creation 101
Activating an Upgrade Kit 102
Controlling the Order of Upgrade Kit Installation 102
Deactivating an Upgrade Kit 103
Applying an Upgrade Kit 104
Converting an Optional Upgrade Kit to a Required Upgrade Kit 105
Distributing Upgrade Kits 106
Disabling Siebel Anywhere 110

7 Performing Additional Siebel Anywhere Tasks 113

Performing Additional Siebel Anywhere Tasks 113

Distributing Additional Language Packs 113
Example of Global Deployment with Siebel Anywhere 113
Example of Creating Multiple Upgrade Kits of the Same Type 116
Process of Updating the Siebel Database Schema 117
Deleting Old Upgrade Kits 125
Testing the Consistency of Upgrade Kit Components 126
Creating Upgrade Kits That Check DLL File Versions 127
Example of Constructing a Third Party Software Upgrade Kit 128

8 Troubleshooting for Siebel Anywhere 131

Troubleshooting for Siebel Anywhere 131
Upgrade Kit Wizard Error at Finish 131
New Upgrade Kit Status: Request Submitted 131
New Upgrade Kit Status: In Progress 133
Unable to Download Upgrade Kit 133
Required Upgrade Kit Cannot Be Installed 134
Failed Upgrade Kit Locks the Siebel Administrator's Account 135
Converting a Required Upgrade Kit to an Optional Upgrade Kit 136
Changing Version Settings After Distribution 137

Preface

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

Using Oracle Applications

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

Documentation Accessibility

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

Contacting Oracle

Access to Oracle Support

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

Comments and Suggestions

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

1 What's New in This Release

What's New in Siebel Anywhere Administration Guide, Siebel CRM 20.1 Update

No new features have been added to this guide for this release. This guide has been updated to reflect only product name changes.

What's New in Siebel Anywhere Administration Guide, Siebel CRM 19.1 Update

No new features have been added to this guide for this release. This guide has been updated to reflect only product name changes.

2 Overview of Siebel Anywhere

Overview of Siebel Anywhere

This chapter provides an overview of the capabilities of Siebel Anywhere. It includes the following topics:

- *Features and Benefits of Siebel Anywhere*
- *Roadmap for Using Siebel Anywhere*
- *About the Siebel Anywhere User Interface Features*
- *About Siebel Anywhere File Attachments*
- *About Siebel Anywhere Version Settings*
- *Siebel Anywhere Terminology*
- *Siebel Anywhere Deployment Recommendations*

Features and Benefits of Siebel Anywhere

Siebel Anywhere allows the Siebel administrator to distribute software upgrades (in the form of Siebel Anywhere upgrade kits) to Siebel Mobile Web Clients, Siebel Developer Web Clients, and Siebel Servers. Upgrades can include custom configurations (updated CFG files or Siebel runtime repositories), certain types of Siebel Business Applications releases, custom extensions to the Siebel database schema, custom files, or third-party files or applications.

Siebel Anywhere provides the following mechanisms:

- Configuration control to make sure that users are connecting to a system with the appropriate software components.
- Creation and packaging of supported types of upgrade components for distribution to clients or servers.
- Distribution to provide users with acceptable version levels of the upgrade components.

> **Note:** Siebel Anywhere is not designed to deliver new Siebel software installations, deliver upgrade kits to Siebel Tools, or support database upgrades of the type described in *Siebel Database Upgrade Guide* . In general, the term upgrade as used in this guide has a specific, narrow meaning relating to distributing software updates through Siebel Anywhere.

For more information about how Siebel administrators and other users work with Siebel Anywhere, see *Roadmap for Using Siebel Anywhere* and other topics in this chapter. For more information about how Siebel Anywhere performs its functions, see *About Siebel Anywhere Version Settings*.

> **Note:** Some of the references in this guide to Siebel CRM version numbers are for example or illustrative purposes. You must verify the availability, characteristics, and requirements of any release. Tasks and requirements for all future releases are subject to change. For more information, see the *Siebel Installation Guide* .

Roadmap for Using Siebel Anywhere

This topic describes the overall process of using Siebel Anywhere.

To use Siebel Anywhere to distribute product updates, perform the following tasks and processes:

1. Plan and prepare to create upgrade kits. For more information about this step, see *Process of Planning and Preparing to Create Upgrade Kits*.

   For example, you must determine your upgrade requirements. As a Siebel administrator, you must create any required configurations, dynamic configuration assignments, and components.

   For a discussion of components and configurations, see *Siebel Anywhere Terminology*.
2. Define an upgrade kit to meet your requirements.

   For a discussion of upgrade kits, see *Siebel Anywhere Terminology*. For more information about this step, see *Defining Upgrade Kits*
3. Activate the upgrade kit. Activating an upgrade kit gathers the files to be included in the upgrade kit and compresses them into a single archive stored on the Siebel File System.

   For more information about this step, see *Activating an Upgrade Kit*.
4. Apply the upgrade kit. Applying an upgrade kit updates a compiled information string with the component version information.

   For more information about this step, see *Applying an Upgrade Kit*.
5. (Client upgrade kits only) Distribute the upgrade kit to test users for testing. Test users are specified by membership in a test configuration.

   For more information about this step, see *Distributing Upgrade Kits*.
6. (Client upgrade kits only) For Siebel client users belonging to the test configuration, these users retrieve the upgrade kit for testing:
   - Mobile Web Client users retrieve the upgrade kit by synchronizing (if the upgrade kit is required) or by selecting and upgrading the applicable component in the Component Upgrades view in the User Preferences screen (if the upgrade kit is optional).

     > **Note:** To support the Component Upgrades view, Siebel administrators must include this view in the responsibilities that are assigned to mobile users. For more information, see *Retrieving and Installing Upgrade Kits*.

   - Where applicable, Developer Web Client users are automatically prompted to retrieve the upgrade kit upon logging in to the Siebel application.

     > **Note:** As part of your planning process, see also *Performing Additional Siebel Anywhere Tasks* which has additional information for particular kinds of upgrades.

   For more information about this step, see *Retrieving, Installing, and Testing Upgrade Kits*
7. (Client upgrade kits only) As a Siebel administrator, correct any problems you discovered during testing. Repeat client testing as necessary.

   For more information, see *Troubleshooting for Siebel Anywhere*
8. (Client upgrade kits only) Use additional configurations to distribute the upgrade kit for wider use, and notify the appropriate users that it is available.

9. (Client upgrade kits only) All applicable Siebel client users retrieve the upgrade kit through the same mechanisms.

About the Siebel Anywhere User Interface Features

This topic describes the user interface features for Siebel Anywhere. It contains the following information:

- *Siebel Anywhere Screens and Views*
- *Siebel Anywhere Wizards and Utilities*

Siebel Anywhere Screens and Views

This topic identifies the screens and views that you use with Siebel Anywhere.

Most of the administrative tasks concerned with Siebel Anywhere are done by using views in the Administration - Siebel Anywhere screen. You must be logged in to the Siebel application as a Siebel administrator to access the views in this screen and perform Siebel Anywhere administration tasks.

This topic is part of *About the Siebel Anywhere User Interface Features*.

When you install the Siebel Business Applications, you must also create a Siebel administrator account that can be used to perform the tasks described in this guide. For more information, see the *Siebel Installation Guide* .

Siebel Anywhere client users use the Component Upgrades view of the User Preferences screen for retrieving and installing optional upgrade kits. This view also allows users to see their current update status. The Component Upgrades view is available only for the Mobile Web Client and Developer Web Client.

> **Note:** To support the Component Upgrades view, Siebel administrators must include this view in the responsibilities that are assigned to mobile users. See *Retrieving and Installing Upgrade Kits*.

This guide provides additional information about these screens and views in later chapters, as part of the information that accompanies specific procedures.

Siebel Anywhere Wizards and Utilities

This topic briefly describes the following Siebel Anywhere wizards and utilities, which are used during various stages of upgrades involving Siebel Anywhere:

- Upgrade Kit Wizard
- Upgrade Kit Builder
- Upgrade Wizard

> **Note:** The tasks and wizards described in this book are not related to the tasks or wizards described in *Siebel Database Upgrade Guide* . In general, the term *upgrade* as used in this guide has a specific, narrow meaning relating to the functions provided by Siebel Anywhere.

This topic is part of *About the Siebel Anywhere User Interface Features*.

Upgrade Kit Wizard

The Upgrade Kit Wizard is the task-oriented feature that Siebel administrators must use to define an upgrade kit. The Upgrade Kit Wizard uses a browser-based user interface to prompt the administrator for the required information. The Upgrade Kit Wizard performs the following functions:

- Allows the Siebel administrator to provide input to define the upgrade kit. The user interface varies according to the component selected for the upgrade kit.
- Validates the parameters supplied by the administrator.
- Handles file attachments for certain types of upgrade kits.
- Specifies upgrade kit items.
- Specifies the sequence of events in the upgrade being defined.
- Submits a schedule-mode server request to invoke the Upgrade Kit Builder when kit definition is complete.

For more information about using the Upgrade Kit Wizard, see *Defining Upgrade Kits*

> **Note:** Do not confuse the Upgrade *Kit* Wizard with the Upgrade Wizard, which is described later in this topic. The Upgrade *Kit* Wizard constructs the upgrade kit. The Upgrade Wizard reads the upgrade kit and installs it on the Siebel client or server.

Upgrade Kit Builder

The Upgrade Kit Builder (alias UpgKitBldr) server component is part of the Siebel Anywhere (alias SiebAnywhere) component group on the Siebel Server. This server component constructs upgrade kits on the server, based upon definitions of upgrade kits collected by the Upgrade Kit Wizard.

During upgrade kit construction, the Siebel administrator can monitor the progress of the Upgrade Kit Builder. To do this, navigate to the Administration - Server Management screen and the Jobs view, and query for Upgrade Kit Builder in the Jobs list. Alternatively, check the Upgrade Kit Builder log file. The associated log file is UpgKitBldr_*taskID*.log.

For more information about enabling the Siebel Anywhere component group, see *Siebel Anywhere Deployment Recommendations*.

Upgrade Wizard

The Siebel Upgrade Wizard (generally referred to here as the Upgrade Wizard) is a stand-alone executable program that reads and installs upgrade kits on Siebel Mobile or Developer Web Clients or Siebel Servers. The Upgrade Wizard also tracks the upgrade process and automatically recovers from errors. In the Windows environment, the associated log file is UpgWiz.log. In the UNIX environment, the associated log file is srvrupgwiz1.log.

> **Note:** Do not confuse the Upgrade Wizard with the Upgrade *Kit* Wizard, which is described earlier in this topic. The Upgrade *Kit* Wizard constructs the upgrade kit. The Upgrade Wizard reads the upgrade kit and installs it on the Siebel client or server.

During the installation of an upgrade kit for a Siebel client, the Upgrade Wizard creates a backup of affected files in the or upgrade folders. If an error occurs during the upgrade, then the Upgrade Wizard attempts to roll back the changes and restore the computer to its original state. Subsequently, when the user starts the Siebel client, the Siebel application detects that an upgrade is either in progress or has failed and notifies the user.

The Upgrade Wizard cannot roll back to the previous version after the upgrade has been installed successfully on a Siebel client. After a successful upgrade, the Upgrade Wizard deletes the backup files, and restoring to a previous configuration is not possible. However, you might be able to roll back from the latest Siebel Innovation Pack release (Siebel CRM version 17.0) to a previously installed Siebel Innovation Pack release (such as Siebel CRM version 8.1.1.14), by using the method described in the uninstallation chapter of the *Siebel Installation Guide* .

About Siebel Anywhere File Attachments

This topic briefly describes how Siebel Anywhere works with file attachments.

Siebel Anywhere stores two kinds of file attachments in the Siebel File System:

- **Individual upgrade kit item files.** Upgrade kit item files are files that are included when a particular upgrade kit is built. Examples include configuration files (CFG files), which are used for configuration upgrade kits, and Siebel runtime repository data files (DAT files), which are used for Siebel Repository Export Import upgrade kits. Depending on the component types being upgraded, an upgrade kit might need to include zero, one, or multiple upgrade kit item files.

  Upgrade kit item files are stored in the Siebel File System after you click Finish in the Upgrade Kit Wizard. The files are visible in the Siebel File System as soon as the status of the upgrade kit record is Pending in the Upgrade Kits view (in the Administration - Siebel Anywhere screen). In the Siebel File System, individual upgrade kit item files are assigned file names that have the format S_UPG_KIT_IARG_*ROWID_REVNO*.SAF.

  In Siebel Anywhere file attachment names, *ROWID* is a unique number combination that identifies the upgrade kit record in the database, and *REVNO* indicates whether the upgrade kit has been revised. If you deactivate and reactivate an upgrade kit, then a new complete upgrade kit file is created, using the same *ROWID* value but a different *REVNO* value in the file name.

  The file name extension SAF is used for all compressed files stored in the Siebel File System, including Siebel Anywhere file attachments.
- **Complete upgrade kit files.** Upgrade kit files are the files that end users will download. Each upgrade kit file contains instructions for updating a specific Siebel component. Some upgrade kit files can also contain one or more upgrade kit item files, as described earlier in this topic. If upgrade kit item files are specified in the Upgrade Kit Wizard, then the specified files are incorporated into the complete upgrade kit file when the upgrade kit is built.

  Complete upgrade kit files are stored in the Siebel File System after you activate the upgrade kit. In the Siebel File System, complete upgrade kit files are assigned file names that have the format S_UPG_KIT_*ROWID_REVNO*.SAF, following the same conventions described for individual upgrade kit item files.

About Siebel Anywhere Version Settings

This topic describes how Siebel Anywhere uses version settings to perform its functions, beginning with some brief definitions of essential terminology.

Siebel Anywhere packages and delivers certain kinds of software using special files called upgrade kits. For more information about upgrade kits, see *Upgrade Kits*.

A software module that is upgraded as a single unit is called a Siebel Anywhere component. Examples of components include those for Siebel configuration files, Siebel database schemas, Siebel client executable programs for Siebel Innovation Pack releases, Siebel Patchset releases (which incorporate multiple quick fixes), Siebel runtime repositories, third-party software, and customer revisions. Any component that you are upgrading using Siebel Anywhere must have its own upgrade kit. For more information about Siebel Anywhere components, see *Upgrade Components*.

A Siebel client user that has been associated with one or more Siebel Anywhere components is called a Siebel Anywhere subscriber. The association between a subscriber and a set of components is not direct; the association is formed by means of the subscriber's membership in an upgrade configuration, which is a definition of a setup used by a particular group of users, such as Siebel Call Center clients or Siebel Sales clients. A configuration associates a group of subscribers with the specific set of upgrade components that those subscribers must have managed and maintained.

For more information about configurations, see *Upgrade Configurations*. For more information about subscribers, see *Siebel Anywhere Subscribers*.

Siebel Anywhere stores and checks several kinds of version information to determine whether a particular subscriber can or must use a particular upgrade kit. To create upgrade kits that have the effects you want, you must understand how these versions are specified, stored, and used.

The following briefly describe the kinds of version information that Siebel Anywhere uses and how Siebel Anywhere uses them:

- *About Specifying a New Version Number for a Component*
- *About Specifying the Versions That Can Use the Upgrade Kit*
- *About Specifying Acceptable Version Ranges for Upgrade Components*
- *About Storing Version Information*
- *How Siebel Anywhere Component Versions Are Used*

About Specifying a New Version Number for a Component

A Siebel administrator who creates an upgrade kit specifies the version number that the component being upgraded will have after the upgrade kit is installed. This version number is called the New Version number.

For example, if the administrator sets the value of New Version to 3 when creating a new customer revision upgrade kit, then a Mobile Web Client on which an upgrade kit successfully installs is upgraded to version 3 of the customer revision component.

About Specifying the Versions That Can Use the Upgrade Kit

The Siebel administrator who creates an upgrade kit also specifies which previous versions of the component are compatible with the upgrade kit. If one or more specific versions are specified, then a client or server must already have one of those versions to download and use the upgrade kit.

The following Upgrade Kit Wizard settings are used to specify the acceptable range of versions for downloading the upgrade kit:

- Minimum Old Version
- Maximum Old Version

These settings are specified as part of the process of defining the upgrade kit. For more information about defining upgrade kits, see *Defining Upgrade Kits*.

As an example, suppose you assign the following values when creating a customer revision upgrade kit. (These values might be suitable when New Version is set to 3 for the upgrade kit.)

- Minimum Old Version = 1
- Maximum Old Version = 2

These example values cause the following results:

- Mobile Web Clients with version 0 of the customer revision component cannot download the upgrade kit until they execute an upgrade kit that changes their customer revision number from 0 to either 1 or 2.
- Mobile Web Clients with version 1 or 2 of the customer revision component can download the upgrade kit.

As a second example, suppose you assign the following values when creating an upgrade kit for a custom component:

- Minimum Old Version = NULL
- Maximum Old Version = NULL

Null values for both of these settings indicate that subscribers who have any previous version of the component, or no previous version of the component, can download and use the upgrade kit. Use null values for these settings only if there are no prerequisite versions for the component, or if the upgrade kit will contain all prerequisites within itself.

As a third example, suppose you want to distribute two upgrade kits for the same custom component, such as a report and a batch file that will manipulate the report. To make sure that the report is installed before the batch file runs, you would create one upgrade kit for the report and a second upgrade kit for the batch file, and you would make the upgrade kit for the batch file dependent on the upgrade kit for the report. The following table lists the settings that would accomplish this objective.

Upgrade Kit Description	Minimum Old Version Value	Maximum Old Version Value	New Version Value
Report (upgrade kit #1)	NULL	NULL	1
Batch file to manipulate the report (upgrade kit #2)	1	1	2

About Specifying Acceptable Version Ranges for Upgrade Components

Siebel Anywhere also stores the acceptable range of versions for each upgrade component. For each subscriber, Siebel Anywhere checks the following:

- Which upgrade component versions are acceptable for running the Siebel application
- Which components must be upgraded

The acceptable range of versions is defined by the Min Version and Max Version settings. Min Version specifies the earliest acceptable version for a component. A component must be upgraded if its version number is less than the value of Min Version. A component does not require upgrading if its version number falls between the values of Min Version and Max Version, inclusive.

Min Version and Max Version values are assigned automatically when you apply an upgrade kit (that is, when you update a compiled information string with the component version information, before distributing the upgrade kit).

There are two ways to apply an upgrade kit. For more information about applying an upgrade kit while using the Upgrade Kit Wizard to define the upgrade kit, see *Defining Upgrade Kits* For more information about applying an upgrade kit using the Apply Upgrade Kit Version Information dialog box, see *Applying an Upgrade Kit*.

Regardless of when the upgrade kit is applied, the values that are assigned to Min Version and Max Version are as follows:

- For each upgrade kit, the default value for Min Version is set to the current Min Version value for the component to be upgraded. This value makes the upgrade kit optional for all users who have at least that version of the component.

  If you prefer to make the upgrade kit required for all users, then you can do so by performing either of the following actions:

  - Select the Required Upgrade Kit check box in the Upgrade Kit Wizard
  - Click the Require Upgrade Kit button in the Apply Upgrade Kit Version Information dialog box.

    Either of these actions sets Min Version for the upgrade kit to match the value of New Version for the upgrade kit.
- The value for Max Version for an upgrade kit always matches the value of New Version for the upgrade kit.

As an example, suppose the following values are automatically assigned when you use the Apply Upgrade Kit Version Information dialog box for a customer revision upgrade kit:

- Min Version = 2
- Max Version = 4

These values cause the following results:

- The upgrade kit is required for any Mobile Web Client with version 1 of the customer revision component. If users have a version less than the minimum and choose not to execute the upgrade kit, then they can only access the application in a read-only mode.
- The upgrade kit is optional for any Mobile Web Client with version 2, 3, or 4 of the customer revision component. Users with these versions can either execute the upgrade kit or continue using their current version of the component.

About Storing Version Information

When a Siebel administrator applies an upgrade kit by using the Apply Upgrade Kit Version Information dialog box, the following settings for the upgrade kit are recorded in the form of a compiled information string in the Siebel database:

- New Version
- Minimum Old Version

- Maximum Old Version
- Min Version
- Max Version

How Siebel Anywhere Component Versions Are Used

After Siebel Anywhere has stored an upgrade kit's version information in the database and the Siebel administrator has distributed the upgrade kit, eligible subscribers have access to the upgrade kit. Siebel Anywhere compares the subscriber's current component version with the version information in the database under conditions that depend on both the type of subscriber and certain settings. For detailed information about when these version checks are done, see the following table.

Subscriber Type	When Version Check Occurs
Mobile Web Client and Developer Web Client users	Mobile and Developer Web Clients perform the version check when the client starts up and connects to either the server or local database. Mobile Web Clients also perform a version check during each synchronization session.
Regional Node Servers	Performed as the Regional Node Server starts up, dependent upon the version check flag on the Regional Node Server. The Replication Agent on a Regional Node Server also performs a version check for each synchronization session, dependent upon the same Version Check parameter.

Multiple factors affect what happens after Siebel Anywhere performs a version check. These factors include the subscriber type, the state the Siebel application is in when the versions are compared, and the relative version numbers.

In general, if a version check reveals that a subscriber is required to upgrade, then that subscriber is prompted to do so and has limited or no access to the affected application until the upgrade is complete. For detailed information about responses to the version check process when the upgrade is required, see *About Specifying the Versions That Can Use the Upgrade Kit*.

However, if a version check reveals that a subscriber is *not* required to upgrade, then that subscriber generally is not prompted to upgrade, but can voluntarily navigate to the Component Upgrades view in the User Preferences screen to discover whether an upgrade is available and to request the upgrade. For more information about displaying optional upgrade kits and requesting optional upgrades, see the following table.

Client or Server	Status of Application: Running	Status of Application: Startup
Mobile Web Client	During synchronization, if a version check discovers a required upgrade kit, then a prompt appears asking if the user wants to download the upgrade kit now. The choices are: *Yes.* The upgrade takes place. *No.* Synchronization does not occur.	If a version check discovers a required upgrade at application startup, then a message such as the following appears: *Client is being started in read-only mode. Please start a Siebel Remote session to retrieve the upgrade kit required to upgrade your system.*

Client or Server	Status of Application: Running	Status of Application: Startup
	If you restart synchronization prior to exiting from the application, then you will be prompted again.	
Developer Web Client	Not applicable.	If version check discovers a required upgrade at application startup, then a prompt appears asking if the user wants to download the upgrade kit now. The choices are: *Yes.* The upgrade takes place. *No.* The application starts in read-only mode.
Regional Node Server	If Version Check is TRUE, then Replication Agent automatically downloads the upgrade kit and shuts down the Regional Node Server. If Version Check is FALSE, then Replication Agent stops merging but does not shut down the Regional Node Server.	If Version Check is set to TRUE, then the Regional Node Server downloads the upgrade kit and shuts down. The Siebel administrator must invoke the Upgrade Wizard manually, from the command line, to perform the upgrade. If Version Check is FALSE, then the Regional Node Server starts up.

Siebel Anywhere Terminology

This topic provides information about the following crucial terms and concepts for understanding and working with Siebel Anywhere:

- *Upgrade Kits*
- *Upgrade Components*
- *Upgrade Configurations*
- *Siebel Anywhere Subscribers*

Upgrade Kits

A Siebel Anywhere *upgrade kit* is an archived file that contains software or Siebel database schema changes required to upgrade a specific upgrade component on a subscriber's computer. An upgrade kit contains one or more upgrade kit

items, which are instructions for actions to be performed and the files associated with those actions. An upgrade kit also contains information about the sequence in which the actions are to be performed.

The available actions include:

- Executing any type of program or batch script. For example, this could include installing a Siebel Innovation Pack release or a Siebel Patchset release.
- Copying, moving, or deleting files.

> **Note:** In this guide, these actions are generally referred to as *installing* the upgrade kit, even where some actions might not represent installation in the same sense as what is described in the *Siebel Installation Guide* or in Siebel Patchset documentation.

The Siebel Anywhere architecture supports the creation of upgrade kits in a Web deployment. A Siebel administrator can use a Web browser, without any locally installed Siebel Business Applications software, to perform many Siebel Anywhere administrative tasks. The administrator prepares upgrade kits by using the Upgrade Kit Wizard and the Upgrade Kit Builder server component.

After the Siebel administrator creates an upgrade kit, it is automatically stored on the Siebel File System. From this location, it is available for retrieval and installation by subscribers. Files or scripts that are included in the upgrade kit are stored in compressed form. For more information about how Siebel Anywhere stores and identifies upgrade kits and the files that are included in upgrade kits, see *About Siebel Anywhere File Attachments*.

It is very important to test upgrade kits. For example, it is recommended that you distribute each upgrade kit to selected mobile users through the use of a test configuration, and have those users attempt to download and execute the upgrade kit before you distribute the upgrade kit to a wider group of users.

Upgrade kits for Siebel client subscribers can be either required or optional.

Required Upgrade Kits

A *required upgrade kit* is an upgrade kit that must be retrieved and installed on the intended Mobile or Developer Web Clients or Siebel Servers. The Siebel clients and servers will prompt the user to upgrade if there is a required upgrade. If a required upgrade is not implemented, then the Siebel client can only be started in read-only mode and the Siebel Server will not start.

Upgrade kits created for Siebel Servers must always be defined as required upgrade kits.

Optional Upgrade Kits

An *optional upgrade kit* is an upgrade kit that is made available to subscribers, but subscribers are not required to retrieve and execute the upgrade kit. For example, optional upgrade kits are useful for making third-party applications, files, or other noncritical components available for users to retrieve at their discretion.

Siebel Anywhere does not automatically prompt users to retrieve and execute optional upgrade kits. Optional upgrade kits are manually retrieved using the Component Upgrades view (in the User Preferences screen), and are installed using the Upgrade Wizard.

> **Note:** To support this functionality, Siebel administrators must include the Component Upgrades view in the responsibilities that are assigned to mobile users. See *Retrieving and Installing Upgrade Kits*.

CAUTION: It is strongly recommended that you use the optional upgrade kit method when testing *every* Siebel Anywhere component upgrade. When an upgrade kit is created as an optional upgrade kit, test users can retrieve the upgrade kit from the Component Upgrades view whenever it is convenient to do so. If the upgrade kit is created as a required upgrade kit, then test users can retain read and write access to Siebel Business Applications only if they upgrade when they are automatically prompted to do so. After testing, you can make the upgrade kit required or optional.

Upgrade Components

An *upgrade component* is a logical unit of software for which Siebel Anywhere performs version checks to determine whether that software must be upgraded. The determination is made by comparing the subscriber's existing version of the component with the version requirements specified in an upgrade kit. Each upgrade component defines how to check versions for a particular software module. For example, the upgrade component Siebel Sales CFG, which is used to check the version of the Siebel Sales client configuration file (siebel.cfg), defines how to locate the file and how to read the version from it. For more information about how Siebel Anywhere conducts version checks, see *About Siebel Anywhere Version Settings*.

Upgrade components can be thought of as belonging to two broad categories:

- *History-dependent* components depend on the presence of certain previously installed software.
- *History-independent* components can be upgraded regardless of the software previously installed.

Components are associated with subscribers by means of Siebel Anywhere configurations, which are discussed in more detail later in this topic. A configuration contains one or more upgrade components. The Siebel administrator can include one or more upgrade components in a configuration.

Siebel Business Applications provide several preconfigured upgrade components. These components are available based upon the Siebel languages for which seed data has been added to the Siebel database. The Siebel administrator can also create additional Siebel Anywhere components or other third-party software. Siebel Anywhere can upgrade these components, just as it does the predefined components used by Siebel Business Applications. For example, it would be possible to create an upgrade component to run a batch program in the form of a BAT or EXE file.

Note: Make sure that all of the components that you need are included in all applicable Siebel Anywhere configurations before you try to deliver upgrade kits for those components.

Note: It is recommended that you keep only one upgrade kit for history-independent components. Depending on your preference, you can either delete or deactivate previous upgrade kits for a history-independent component. Deleting an upgrade kit increases available space in the Siebel File System. Deactivating an upgrade kit prevents use of the upgrade kit, while keeping it available in case you need it unexpectedly.

Each upgrade component is classified by a component type, as described in the following table. For more information about upgrade components, see *Setting Up Custom Siebel Anywhere Upgrade Components*.

Siebel Component Type	Siebel Anywhere Component	Comments
Customer Revision	Siebel Client Customer Revisions Siebel Client Customer Revisions__ *language-code*	This nonspecific component distributes any customer-revised files, such as Online Help files, Siebel Help files, Siebel Sample Database files, or third-party files or software on the Siebel client. Can be either history-dependent or history-independent.
Customer Revision	Siebel Server Customer Revisions Siebel Server Customer Revisions_ _*language-code*	This nonspecific component distributes any customer-revised files, such as Online Help files, Siebel Help files, Siebel Sample Database files, or third-party files or software on the Siebel client. Can be either history-dependent or history-independent.
Siebel Upgrade Wizard	Siebel Upgrade Wizard	This component distributes the Upgrade Wizard. It is automatically associated with every upgrade kit to make sure that the client uses the current version of the wizard. History-independent.
Siebel Database Extension	Siebel Database Schema	This component distributes updates for the Siebel database schema used by the Siebel Regional Node database and by Siebel Remote mobile databases. History-independent.
Siebel Executables	Siebel Client Executables Siebel Client Executables__ *language-code*	This component distributes files for installing Siebel Innovation Pack releases. History-independent. Use the Siebel Client Executables components for creating Siebel Innovation Pack upgrade kits.
Siebel Quick Fix	Siebel Quick Fix	This component distributes files for installing Siebel Patchset releases. Use the Siebel Quick Fix component for creating Siebel Patchset upgrade kits.
Siebel Repository ExportImport	Siebel Repository Export Import	This component distributes files for Siebel runtime repository updates to Mobile Web Clients. History-dependent. Use the Siebel Repository Export Import component for creating Siebel runtime repository upgrade kits.

Siebel Component Type	Siebel Anywhere Component	Comments
Third Party Software	Third Party - Adobe Acrobat Third Party - Adobe Acrobat Reader Third Party - IBM DB2 Client Third Party - Microsoft Internet Explorer Third Party - Microsoft SQL Server Driver Third Party - Microsoft Word Third Party - Oracle 8 Client (use for any supported Oracle version)	This component distributes third-party software, as used on the Siebel client. Can be either history-dependent or history-independent. Some Third Party Software upgrade components are for version checking only. Siebel Anywhere is not intended to deliver complex third-party software products by using upgrade kits. For smaller and simpler software products such as Adobe Acrobat and WinZip, you can use Siebel Anywhere to deliver them as Third Party Software upgrade kits. **Note:** Before using any Third Party Software components, update Min Version values to conform to supported versions noted in the Certifications tab on My Oracle Support.

Upgrade Configurations

An upgrade configuration is a definition of the setup used by a particular set of Siebel Anywhere subscribers, such as users of Siebel Call Center clients or Siebel Sales clients. A configuration associates a particular set of Siebel Anywhere subscribers with the specific set of upgrade components that those subscribers must have managed and maintained. Each Siebel subscriber belongs to an individual Siebel Anywhere configuration.

When Siebel Anywhere checks whether a particular subscriber needs an upgrade, it checks the versions of all components included in that subscriber's configuration.

Siebel Business Applications provide client configurations to support different applications. Some examples include the following:

- Siebel Call Center Client
- Siebel Marketing Client
- Siebel Partner Manager Client
- Siebel Sales Client
- Siebel Service Client
- Siebel Test Client

Each subscriber can be assigned to one of these configurations. Siebel administrators can also create new configurations for special situations or for applications or products for which predefined components or configurations are not provided.

> **Note:** Siebel Anywhere supports global deployments by including seeded upgrade configurations and components for some supported languages included within your Siebel Business Applications. You might need to create new configurations to support languages for which seed data is not provided. Consider carefully the receivers or subscribers to any upgrade kit you create, and use the correct upgrade component based on end-user languages for Siebel applications.

By default, Siebel Anywhere uses the value of the ComponentName entry in the [Siebel] section of a Siebel client application's CFG file to determine what configuration is used by the client user (employee) and thus what components are to be checked. The server parameter Upgrade Component specifies the configuration for Siebel Server.

It is possible to override a client's default configuration assignment by using the Administration - Siebel Anywhere screen to assign the client to a different configuration. This method of assigning a client to a configuration is appropriate for situations in which the assignment is temporary. For example, a client might be assigned to a specific configuration to let that client's user test an upgrade kit for a particular application's CFG file before it is released for general use. These temporary assignments are also called *dynamic assignments*.

For more information about viewing the employees that are dynamically associated with Siebel Anywhere configurations, see *Identifying Configurations to Deliver Upgrade Components*. For more information about modifying and creating Siebel Anywhere configurations, see *Modifying and Creating Siebel Anywhere Configurations*.

Siebel Anywhere Subscribers

A Siebel Anywhere subscriber is a Siebel client or a Siebel Server that belongs to an active configuration. It can be a laptop computer, a desktop computer, or a Regional Node Server. A Siebel Anywhere subscriber belongs to one upgrade configuration. Types of subscribers include the following:

- Mobile Web Clients of Siebel Business Applications such as Siebel Sales, Siebel Partner Manager, Siebel Field Service, or Siebel Call Center
- Developer Web Clients of these applications
- Siebel Servers operating against regional databases (referred to as Regional Node Servers)

For more information about viewing the employees that are dynamically associated with Siebel Anywhere configurations, see *Identifying Configurations to Deliver Upgrade Components*.

> **CAUTION:** It is strongly recommended that you run *only* the Siebel Web Client (the client type that does not require a Siebel installer) for user accounts that have Siebel administrator responsibilities, to make sure that administrative tasks are performed while connected to the HQ Server, and to make sure that the administrator is not prevented from logging in for reasons related to component versions. If you run the Siebel Developer Web Client for any administrator account, then it is strongly recommended that the account *not* be associated with a Siebel Anywhere configuration. This precaution also helps prevent version-related login problems.

Siebel Anywhere Deployment Recommendations

In order for you to be able to create upgrade kits, the Siebel Anywhere component group must be enabled on a Siebel Server and the Upgrade Kit Builder server component must be running.

> **Note:** When initially configuring a Siebel Server, do not enable the Siebel Anywhere component group. Wait until every Siebel Server is installed and configured, and then enable the Siebel Anywhere component group on only one Siebel Server.

> **CAUTION:** Do not enable the Siebel Anywhere component group on a Regional Node Server. The reason for this is that upgrade kits must be built sequentially in the correct order. If the Upgrade Kit Builder server component is enabled on multiple servers, then upgrade kits might be built in the incorrect order, because multiple upgrade kits can be created at the same time.

For details about how to enable a component group on a Siebel Server, see *Siebel System Administration Guide* . See also the *Siebel Installation Guide* .

3 Planning and Preparing to Use Siebel Anywhere

Planning and Preparing to Use Siebel Anywhere

This chapter describes planning and other preliminary tasks that must be completed before you use Siebel Anywhere to create and distribute an upgrade kit. It includes the following topics:

- *Process of Planning and Preparing to Create Upgrade Kits*
- *Determining Your Upgrade Requirements*
- *Modifying and Creating Siebel Anywhere Configurations*
- *Setting Up Custom Siebel Anywhere Upgrade Components*
- *Verifying Client Connections to the Siebel File System*
- *Verifying the Availability of Siebel Anywhere*
- *Preparing Upgrade Kit Contents*
- *Creating a Delta Patch File*
- *Synchronizing Server Components*

Process of Planning and Preparing to Create Upgrade Kits

This topic provides an overview of how to plan a Siebel Anywhere upgrade and complete the tasks that must precede creation of one or more upgrade kits.

To plan and prepare to create upgrade kits, perform the following tasks:

1. *Determining Your Upgrade Requirements*. Gather information about the software to be distributed and the computers that will receive it. The information you gather determines how many components will be involved and how many upgrade kits will be needed, among other important points.
2. *Modifying and Creating Siebel Anywhere Configurations*. Determine whether existing Siebel Anywhere configurations match your upgrade needs and then create or modify any configurations you will need.
3. *Setting Up Custom Siebel Anywhere Upgrade Components*. Determine whether existing Siebel Anywhere components match your upgrade needs and then create any custom components you will need.
4. *Verifying Client Connections to the Siebel File System*. Verify settings for connections to the Siebel File System for your clients and servers.
5. *Verifying the Availability of Siebel Anywhere*. Verify the availability of the Siebel Anywhere component group on the Siebel Server.
6. *Preparing Upgrade Kit Contents*. Gather (and, where necessary, create) the files to be included in each upgrade kit. The specific process to follow depends on the types of components you are planning to upgrade.

> **Note:** Some upgrade kits do not require any files. For example, Siebel Database Schema upgrade kits do not require any files to create an upgrade kit.

When these parts of the process have been completed, you can use the information, infrastructure elements, and files you have prepared to define the upgrade kits that you need. This process is described in *Defining Upgrade Kits*.

> **CAUTION:** Any errors in an upgrade kit, whether in the files being distributed or in application processes defined for them, can have widespread impact on your production environment. To prevent such errors, thorough planning, preparation, and testing of upgrade kits are essential.

Determining Your Upgrade Requirements

Before using Siebel Anywhere to upgrade existing software or distribute new software, you must gather detailed information about the software to be distributed and about the computers that will receive it. This task is a step in *Planning and Preparing to Use Siebel Anywhere*.

Specifically, you must collect the following kinds of information, each of which is described in more detail in the following topics:

- **What software must be replaced or added?** See *Identifying Software to Be Replaced or Added*.
- **Is Siebel Anywhere a suitable method for upgrading or adding this software?** See *Evaluating Siebel Anywhere as an Upgrade or Delivery Method*.
- **Which computers need the software?** See *Identifying Computers and Users to Receive Upgrades*.
- **Which configurations can deliver the necessary components to the designated computers?** See *Identifying Configurations to Deliver Upgrade Components*.
- **What files must each component's upgrade kit contain?** See *Identifying Files to Include in Upgrade Kits*.
- **Will each upgrade kit be required or optional?** See *Determining Whether an Upgrade Kit Is Required or Optional*.
- **What version settings must each component's upgrade kit have?** See *Determining Version-Setting Values*.
- **What specific preliminary tasks must be performed for particular upgrade kit types?** See *Other Preliminary Tasks for Upgrade Kits*.
- **Must multiple components be delivered in a prescribed sequence?** See *Determining Required Upgrade Sequences*.
- **How will you test each upgrade kit before distributing it to end users?** See *Planning Upgrade Test Details*.

Identifying Software to Be Replaced or Added

Siebel Anywhere can upgrade or distribute many kinds of software. The nature of the software to be upgraded or distributed affects many parts of the process of using Siebel Anywhere:

- **Upgrades of existing software.** For upgrades of existing software, you must determine the version numbers used by Siebel Anywhere for previous versions of the software being upgraded. You must also decide whether to allow any of those previous versions to remain in use, or whether to require that they be upgraded.

- **Distribution of new software.** For distribution of new software, you might need to create one or more custom components. Depending on who will use the software, you might also need to create new configurations.

This topic is part of *Determining Your Upgrade Requirements*.

For language-dependent components, planning the upgrade also includes identifying every end-user language used in your Siebel Business Applications implementation. For example, a CFG file associated with a specific language is a language-dependent component, such as Siebel Sales CFG__ENU for U.S. English, Siebel Sales CFG__DEU for German, or Siebel Sales CFG__JPN for Japanese. Make sure to use the correct end-user language pack (or upgrade component) for each language included in your Siebel Business Applications while creating upgrade kits. For more information regarding the use of different languages, see *Example of Global Deployment with Siebel Anywhere*.

The number and type of upgrade kits required for your upgrade depend on your subscribers and the components to be upgraded. For example, you might be preparing to upgrade to a new version of your custom Siebel configuration, requiring the distribution of an updated Siebel runtime repository to every Mobile Web Client user. (Siebel runtime repository changes are not distributed by an upgrade kit to Developer Web Clients. Runtime repository changes are visible to Developer Web Clients, without distribution through an upgrade kit, because these clients connect directly to the Siebel database.)

Your custom configuration might also need to apply Siebel database schema changes to your Siebel databases and to the local databases of Mobile Web Clients. Creating a Siebel Database Schema upgrade kit applies the changes from the logical schema to the physical schema. (Siebel database schema changes are not distributed by an upgrade kit to Developer Web Clients. Schema changes are visible to Developer Web Clients, without distribution through an upgrade kit, because these clients connect directly to the Siebel database.)

You must create an upgrade kit for each component to be upgraded. In the preceding example, you would need to create one upgrade kit for the Siebel runtime repository updates and a second upgrade kit for the Siebel database schema extensions. Compile a list of the upgrade kits you will need.

Evaluating Siebel Anywhere as an Upgrade or Delivery Method

Observe the following guidelines:

- Use Siebel Anywhere *only* with software that is related to the use of Siebel Business Applications.
- Do *not* use Siebel Anywhere for major upgrades, such as upgrading a Siebel database schema from Siebel CRM 8.0 to Siebel Innovation Pack 2017 (Siebel CRM version 17.0), or for installing any prior release, such as Siebel CRM 8.1.1 or version 8.2.2.

  > **Note:** It is not necessary to install a prior release such as Siebel CRM version 8.1.1 or version 8.2.2 before you install Siebel Innovation Pack 2017.

This topic is part of *Determining Your Upgrade Requirements*.

Identifying Computers and Users to Receive Upgrades

To make sure that Siebel Anywhere delivers your upgrade kit to the appropriate computers and users, you must take the following steps:

- Determine whether any Siebel Servers need the upgrade. (The steps to follow for servers, and the expected behavior, might differ from the corresponding client steps and behavior.)

- Determine which clients need the upgrade. Some possibilities include:
  - All client users need the upgrade.
  - Client users who have a particular job function need the upgrade.
  - Clients users that have particular prior versions of the software need the upgrade.

This topic is part of *Determining Your Upgrade Requirements*.

Identifying Configurations to Deliver Upgrade Components

A Siebel Anywhere *configuration* is a definition of the setup used by a particular set of Siebel Anywhere subscribers, such as Siebel Call Center clients or Siebel Sales clients. A configuration associates a particular set of Siebel Anywhere subscribers with the specific set of upgrade components that those subscribers must have managed and maintained. Each Siebel subscriber belongs to an individual Siebel Anywhere configuration that includes a specific set of components.

This topic is part of *Determining Your Upgrade Requirements*.

As part of the process of planning your upgrade, you must make sure that your Siebel Business Applications implementation contains configurations that are related to the components you want to upgrade. You must also make sure that those configurations can be used to distribute the upgrade to the appropriate subscribers, whether those subscribers represent servers or clients.

The following procedures provide instructions for listing existing configurations and for inspecting the components and users to which these configurations are related. For more information about Siebel Anywhere configurations, see *Siebel Anywhere Terminology*.

Displaying Siebel Anywhere Configurations and Their Components

The following procedure describes how to display the available Siebel Anywhere configurations and their related components.

To display available Siebel Anywhere configurations and their related components

1. Navigate to the Administration - Siebel Anywhere screen, then the Upgrade Configurations view.
2. In the Upgrade Configurations list, select a configuration that you want to inspect, and then click the Upgrade Components view tab.

   The Upgrade Components list displays the components that are currently associated with the selected configuration. The following table describes some of the fields in this list.

Field Name	Comments
Name	The name of the component. Only single-byte, alphanumeric characters, blank space, underscore, and dash are allowed. Component names cannot include special characters like periods or characters such as slash, asterisk, pipe, question mark, colon, quotes, or angle brackets.
Min Version	The minimum version required for the component on the client system.

Field Name	Comments
	If the client uses a version of the component that is less than the minimum, then the client must upgrade or the application runs in read-only mode. Every component receives automatic version checking.
Max Version	The maximum version allowed for the component on the client system. If clients use a version of the component that is between the minimum and maximum, then they can still use the application in read and write mode without installing an upgrade kit, even if the upgrade kit is otherwise required. Clients can access Siebel Business Applications if their local version is higher than the maximum version for the component. Every component receives automatic version checking.
Min Version Distributed	The version of the component that has been distributed for this configuration. This value is set automatically when upgrade kits for this component are distributed for this configuration.
Max Version Distributed	The version of the component that has been distributed for this configuration. This value is set automatically when upgrade kits for this component are distributed for this configuration.

The following procedure provides instructions for listing the subscribers who are dynamically assigned to a selected configuration. The procedure does not list subscribers who are assigned to a particular configuration by means of a CFG file setting.

Displaying Employees Associated with a Siebel Anywhere Configuration

The following procedure describes how to display the employee users who are associated with a Siebel Anywhere configuration.

To list employees who are dynamically associated with a configuration

1. Navigate to the Administration - Siebel Anywhere screen, then Upgrade Configurations, then the Employees view.
2. In the Upgrade Configurations list, select the configuration for which you want to view the dynamically assigned employees.

   The Employees list displays the employees who are dynamically assigned to the selected configuration. Employees who are assigned to the configuration through their CFG file values are not listed.

If you find that existing configurations do not meet your current upgrade needs, then see *Modifying and Creating Siebel Anywhere Configurations* for information about how to modify existing configurations or create new configurations.

Identifying Files to Include in Upgrade Kits

Most Siebel Anywhere upgrade kits contain software files to be distributed to subscribers. (The exceptions are upgrade kits that contain only instructions to be executed. Such instructions are generated during the upgrade kit creation process, rather than being stored in file attachments.) As part of planning an upgrade kit, identify any files the upgrade kit will include and gather those files into a single directory.

Note: It is recommended that you gather files for only one upgrade kit at a time.

This topic is part of *Determining Your Upgrade Requirements*.

The following table lists some files that are typically included in various upgrade kit types. For more information about the supported component types for upgrade kits, see *Upgrade Components*.

Upgrade Kit Type	Files
Siebel Configuration File (Siebel CFG File component type)	One CFG file for each upgrade kit, such as siebel.cfg (for Siebel Sales), uagent.cfg (for Siebel Call Center), and so on.
Siebel Database Schema (Siebel Database Extension component type)	No files
Siebel Client Executables (Siebel Executables component type)	Files for a Siebel Innovation Pack release.
Siebel Client Executables__*language-code* (Siebel Executables component type)	Files for a Siebel Innovation Pack release that are specifically for a given language.
Siebel Quick Fix (Siebel Quick Fix component type)	Files for a Siebel Patchset release.
Siebel Client Customer Revisions (Customer Revision component type)	Batch files, Web image files, or other files.
Siebel Client Customer Revisions__*language-code*	Message template files.

Upgrade Kit Type	Files
(Customer Revision component type)	
Siebel Repository Export Import (Siebel Repository ExportImport component type)	An update of the Siebel runtime repository, in the form of a data file (DAT) that will be executed in the local database on the Siebel Mobile Web Client.

Determining Whether an Upgrade Kit Is Required or Optional

You can define an upgrade kit as either required or optional, as described in the discussion of upgrade kits in *Siebel Anywhere Terminology*. However, it is recommended that you initially define every upgrade kit as optional. Your test client users can test an optional upgrade kit at their convenience, while keeping read and write access to the Siebel application in the meantime.

When you are satisfied that an optional upgrade kit functions appropriately for test client users, you can modify the upgrade kit to make it required, retest it as a required upgrade kit, and then distribute it to production users as a required upgrade kit. For more information about this process, see *Converting an Optional Upgrade Kit to a Required Upgrade Kit*.

This topic is part of *Determining Your Upgrade Requirements*.

Determining Version-Setting Values

The exact set of information needed for defining a Siebel Anywhere upgrade kit depends on the upgrade kit's component types. However, you must supply version information for all upgrade kits.

When determining who can, or who must, upgrade a particular component, Siebel Anywhere compares version information stored in the database with the versions of components actually installed on each subscriber's computer. The information in the database comes from the information you supply when you create upgrade kits.

This topic is part of *Determining Your Upgrade Requirements*.

For each upgrade kit, you must supply values or approve the default values for the following version settings:

- New Version
- Minimum Old Version
- Maximum Old Version

For a general discussion of the significance of these version numbers, see *About Siebel Anywhere Version Settings*. In general, the values you supply will be related to the version numbers already in use for the component being upgraded. Therefore, gathering version information about existing components is an important part of planning an upgrade.

CAUTION: Make certain that Minimum Old Version, Maximum Old Version, and New Version settings are correctly set before finishing your upgrade kit, using the default numbering system, if possible. Incorrectly specifying the version information can prevent subscribers from upgrading successfully.

Gathering Information About Existing Component Versions

The following procedure provides instructions for listing the components currently available in your Siebel Business Applications implementation, along with information about currently acceptable versions of those components and how those versions are checked.

When implementing upgrades, you must preserve the version numbers for every component. This precaution is important because Siebel Anywhere assigns increasing version numbers, but components that are included in upgrades might have default version numbers set to 0 (zero). If the version number of a new component is lower than the version number of the corresponding preupgrade component, then newly upgraded users might be prompted, incorrectly, to install the old component.

Never reset any Siebel Anywhere version numbers to zero; instead, increase the version numbers of the new components to match their preupgrade counterparts. The version number of your repository is displayed as User Version when you choose Help, then About Repository, in your Siebel application.

To display version information for existing components

- Navigate to the Administration - Siebel Anywhere screen, then the Upgrade Component List view.

  The Upgrade Component List appears.

CAUTION: Do not use the Upgrade Component List to modify information about an existing component. Modifying information in the list can prevent version checking from working properly. If you must modify version information for an existing component, then use the Upgrade Kit Wizard, as described in *Defining Upgrade Kits* If you want to modify other component characteristics, then create a custom component with the characteristics that you need, as described in *Setting Up Custom Siebel Anywhere Upgrade Components*.

The following table describes the information available in the Upgrade Component list.

Field Name	Comments
Component Type	The type of component for upgrade kits; for example, Siebel Repository ExportImport or Third Party Software.
Locate Information	Information used by Siebel Anywhere when locating version information for the subscriber's currently installed component. For more information about how Siebel Anywhere uses this setting, see *About Monitoring and Verifying Siebel Anywhere Version Numbers*.
Locate Method	The method used by Siebel Anywhere to locate version information for the subscriber's currently installed component. For more information about how Siebel Anywhere uses this setting, see *About Monitoring and Verifying Siebel Anywhere Version Numbers*.
Max Version	The latest version of the component that is available for running the application in read and write mode.

Field Name	Comments
Min Version	The earliest version of the component that is acceptable for running the application in read and write mode.
Name	The name that identifies the component for Siebel Anywhere.
Version Information	Information used by Siebel Anywhere when checking the version of the subscriber's currently installed component. For more information about how Siebel Anywhere uses this setting, see *About Monitoring and Verifying Siebel Anywhere Version Numbers*.
Version Method	The method used by Siebel Anywhere to check the version for the subscriber's currently installed component. For more information about how Siebel Anywhere uses this setting, see *About Monitoring and Verifying Siebel Anywhere Version Numbers*.

Planning Version Settings for a New Upgrade Kit

The information you gather about current version settings lets you plan version settings for your new upgrade kit.

Note: When you run the Upgrade Kit Wizard to define your new upgrade kit, you must supply values for the New Version, Minimum Old Version, and Maximum Old Version settings.

The following contains guidelines for determining the version values to use in a variety of situations:

- Choosing your New Version value.
- Choosing your Minimum Old Version and Maximum Old Version values.

You can use these tables as worksheets by printing them out and filling in the values that you will use when you define your upgrade kit for each component.

The New Version value specifies the version number that the component being upgraded will have after the upgrade kit is used. Look in the Situation column (as shown in the following table) to find a description of your situation, and read the adjacent guidelines for setting the appropriate value.

Version Setting	Situation	Value Guidelines	Your Value
New Version (Identifies the version number of the component in the upgrade kit)	New component	Set New Version to 1.	1
New Version (Identifies the version number of the component in the upgrade kit)	Upgrade to existing component	Set New Version to *n* plus 1, where *n* is the Siebel Anywhere version number of the latest version of the component previously installed within your Siebel Business Applications implementation. To display previously installed versions, see *Determining Version-Setting Values*.	___

Version Setting	Situation	Value Guidelines	Your Value
New Version (Identifies the version number of the component in the upgrade kit)	Upgrade to existing component after a major Siebel upgrade	Make sure that any components provided in the major upgrade have Siebel Anywhere version numbers equal to or greater than preupgrade components. You might need to stamp new numbers on the components to achieve this. For more information, see *Testing the Consistency of Upgrade Kit Components*. When component version numbering is continuous for the upgrade, use the instructions for upgrading to an existing component, in the previous row of this table.	___
New Version (Identifies the version number of the component in the upgrade kit)	Replacing a defective upgrade kit that has been distributed and deactivated	Set New Version to the same value you assigned to the defective upgrade kit.	___

The Minimum Old Version and Maximum Old Version values specify the range of component versions that subscribers must have to download and install the upgrade kit. Look in the Situation column (as shown in the following table) to find a description of your situation, and read the adjacent guidelines for setting the appropriate values.

Version Setting	Situation	Value Guidelines	Your Value
Minimum Old Version (Determines who can use the upgrade kit)	No component versions are prerequisites for subscribers using upgrade kit or all needed prerequisites are included in this upgrade kit.	Leave blank.	NULL
Minimum Old Version (Determines who can use the upgrade kit)	Subscribers are upgrading a history-independent component, such as executable, or Siebel database schema.	This setting is automatically set to NULL. It cannot be changed.	NULL
Minimum Old Version (Determines who can use the upgrade kit)	Subscribers are upgrading a CFG file.	It is recommended that you leave this setting blank.	___
Minimum Old Version (Determines who can use the upgrade kit)	Subscribers are upgrading a customer revision component.	It is recommended that you accept the default value for this setting and the setting for Maximum Old Version, to make sure that your users download and install every upgrade kit of this	___

Version Setting	Situation	Value Guidelines	Your Value
		type, without skipping any of them.	
Minimum Old Version (Determines who can use the upgrade kit)	Subscribers are required to have a prior version of the component.	Set to the earliest component version that can download and install the new upgrade kit.	___
Minimum Old Version (Determines who can use the upgrade kit)	You are replacing a defective upgrade kit that has been distributed and deactivated.	Set Minimum Old Version to the same value as you used in the defective upgrade kit.	___
Maximum Old Version (Determines who can use the upgrade kit)	No component versions are prerequisites for subscribers using upgrade kit or all needed prerequisites are included in this upgrade kit.	Leave blank.	NULL
Maximum Old Version (Determines who can use the upgrade kit)	Subscribers are upgrading a history-independent component, such as executable, or Siebel database schema.	This setting is automatically set to NULL. It cannot be changed.	NULL
Maximum Old Version (Determines who can use the upgrade kit)	Subscribers are upgrading a CFG file.	It is recommended that you leave this setting blank.	___
Maximum Old Version (Determines who can use the upgrade kit)	Subscribers are upgrading a customer revision component.	It is recommended that you accept the default value for this setting and the setting for Minimum Old Version, to make sure that your users download and install every upgrade kit of this type, without skipping any of them.	___
Maximum Old Version (Determines who can use the upgrade kit)	Subscribers are required to have a prior version of component.	Set to the latest component version that can download and install the new upgrade kit. This value is always less than the value of New Version.	___
Maximum Old Version (Determines who can use the upgrade kit)	You are replacing a defective upgrade kit that has been distributed and deactivated.	Set Maximum Old Version to the same value as you used in the defective upgrade kit.	___

Other Preliminary Tasks for Upgrade Kits

This topic describes additional preliminary tasks to perform for upgrade kits, based on their specific component types.

This topic is part of *Determining Your Upgrade Requirements*.

Preliminary Tasks for Siebel Database Schema Upgrade Kits

Perform the following tasks before creating a Siebel Database Schema upgrade kit:

- Make sure that the changes to your Siebel database schema can be distributed by this type of upgrade kit.
- Make sure that the ODBC data source correctly points to the server database that has the modified database definition.
- Use SQL queries on your server database to obtain values for the following parameters, as they apply to your database platform:
  - 16K Table Space
  - 32K Table Space
  - Table Group File
- If you are operating in an IBM DB2 environment, then drop all customized views and triggers before you run the Upgrade Kit Wizard to define a Siebel Database Schema upgrade kit. Otherwise, the attempt to create the upgrade kit will fail.

For more information about the overall process of deploying a Siebel Database Schema upgrade kit, see *Process of Updating the Siebel Database Schema*.

Preliminary Tasks for Siebel Repository Export Import Upgrade Kits

Perform the following tasks before creating a Siebel Repository Export Import upgrade kit:

- Using the Siebel Migration application, complete a full runtime repository migration operation.
- Set up the Siebel Mobile Web Client.
- Create a new workspace, make the repository changes, and deliver the updates.
- Using the Siebel Migration application, complete an incremental runtime repository migration operation.
- Make sure that the repository changes are available in the runtime environment to which clients are pointing.
- Make sure that the ODBC data source correctly points to the server database that has the modified database definition.
- If you are running the dataexp.exe directly in the developer environment instead of using the Siebel Migration application, then make sure the workspace is in a check-pointed state.

For more information about planning for a database migration using Siebel Application Interface, preparing data for a migration using Siebel Application Interface, and migrating database data from a source environment to a target environment using Siebel Application Interface, see *Siebel Database Upgrade Guide* .

Preliminary Tasks for Third Party Software Upgrade Kits

Perform the following tasks before creating a Third Party Software upgrade kit:

- Locate or prepare the executable file or script that will be executed on the subscriber's computer when the upgrade kit is installed. Make a note of any input parameters required by the executable file or script for the third-party software.
- Place the executable file in a network location that is accessible to the Siebel Server where the Upgrade Kit Builder is enabled.
- Make a note of the UNC path to the network location where you placed the executable file.
- Determine the location where the third-party software will be installed on the subscriber's computer.

Preliminary Tasks for Siebel Client Customer Revisions Upgrade Kits

Perform the following tasks before creating a Siebel Client Customer Revisions upgrade kit:

- Locate or prepare the executable file or script that will be executed on the subscriber's computer when the upgrade kit is installed. Make a note of any input parameters required by the executable file or script.
- Locate any other files to be included in the upgrade kit, and make a note of the locations where they will be installed on the subscriber's computer.

Determining Required Upgrade Sequences

Sometimes you might need to use more than one upgrade kit to complete an upgrade.

This topic is part of *Determining Your Upgrade Requirements*.

For example, you might want to deliver a Siebel Innovation Pack release, using a Siebel Client Executables upgrade kit, and then deliver an updated configuration file, using another upgrade kit (such as Siebel Sales CFG__ENU) that is dependent on the prior installation of the Siebel Client Executables upgrade kit.

In cases involving multiple upgrade kits, the upgrade kits might need to be installed in a specific sequence. It is possible to automate this process by making one upgrade kit dependent on the prior installation of another upgrade kit. You can specify this dependency before activating the second upgrade kit in the sequence. For more information about how to do this, see *Controlling the Order of Upgrade Kit Installation*.

> **Note:** If you plan to specify a required upgrade sequence, then do not use the Upgrade Kit Wizard to activate the upgrade kits involved.

Planning Upgrade Test Details

It is important to plan your upgrade kit testing ahead of time, to make sure that you create any needed infrastructure elements and choose appropriate settings when you create the upgrade kit.

This topic is part of *Determining Your Upgrade Requirements*.

The following general recommendations are useful for most test plans:

- **Create all upgrade kits as optional upgrade kits.** This facilitates testing by letting users request an upgrade at their convenience. (A required upgrade kit might or might not prompt users to upgrade when they start the application, depending on the version requirements and the test client version.) After you have verified that an optional upgrade kit installs correctly, you can change it to be required, if you want, and retest its installation using another test client.
- **Distribute a new upgrade kit to a test configuration (such as the Siebel Test Client configuration) before distributing it to production configurations.** If necessary, create such a test configuration ahead of time. Make sure that it has the appropriate test employee logins assigned to it, either by means of configuration files or dynamic assignments.
- **Test each upgrade kit with both Mobile Web Clients and Developer Web Clients, as applicable.**

Modifying and Creating Siebel Anywhere Configurations

This topic contains information about modifying and creating Siebel Anywhere configurations. A configuration is a required infrastructure element for the successful creation and deployment of an upgrade kit.

This task is a step in *Process of Planning and Preparing to Create Upgrade Kits*.

This topic contains the following information:

- *Adding Components to a Configuration*
- *Removing Components from a Configuration*
- *Creating a New Configuration*
- *About Assigning Employees to a Configuration*
- *Assigning Employees to a Configuration*
- *Removing Employees from a Configuration*

Note: If a new configuration is created, a configuration is split or merged with another configuration, or the name of an existing configuration is changed, then the ComponentName parameter on the corresponding Siebel clients must also be updated so that the clients can detect any upgrades.

Adding Components to a Configuration

Sometimes you might need to add components to an existing configuration. Any custom component that you create must be added to a configuration before you can use it to distribute upgrades. The following procedure describes how to perform this task.

This topic is part of *Modifying and Creating Siebel Anywhere Configurations*.

To add a component to a configuration

1. Navigate to the Administration - Siebel Anywhere screen, then the Upgrade Configurations view.

2. In the Upgrade Configurations list, select the configuration to which you want to add a component, and then click the Upgrade Components view tab.
3. In the Upgrade Components list, click New.
4. In the Upgrade Components dialog box, double-click the component you want to add.

   The component appears, highlighted, in the Upgrade Components list.
5. Repeat the previous steps for any additional components you want to add to this configuration.

Removing Components from a Configuration

Sometimes you might need to remove components from an existing configuration. The following procedure describes how to perform this task.

This topic is part of *Modifying and Creating Siebel Anywhere Configurations*.

To remove a component from a configuration

1. Navigate to the Administration - Siebel Anywhere screen, then the Upgrade Configurations view.
2. In the Upgrade Configurations list, select the configuration from which you want to remove a component, and then click the Upgrade Components view tab.
3. In the Upgrade Components list, select the component that you want to remove, and then click Delete.
4. In the dialog box, confirm that this record is the record that you want to delete.
5. Repeat the previous steps for any additional components that you want to remove from this configuration.

Creating a New Configuration

In most cases, the configurations provided with Siebel Anywhere are sufficient. However, you can create new configurations for special situations or to represent different installed clients within your company. The procedure in this topic describes how to perform this task.

This topic is part of *Modifying and Creating Siebel Anywhere Configurations*.

For example, if everyone in your company uses Siebel Sales and you must distribute certain upgrades to employees in remote offices separately from the rest of the company, then you would create a new configuration to accommodate this situation. The employees in the remote offices would be temporarily associated with the new configuration so that you could distribute the special upgrade just to them.

For details on associating specific employees with a configuration, See *Assigning Employees to a Configuration*.

Also, you might need to create different configurations based upon language usage. If a subset of the users in your company use only U.S. English (ENU) and another subset uses both U.S. English and German (DEU), then create two different configurations.

If you must create a new configuration for long-term use (as in language usage), then make sure that every client CFG file is appropriately updated.

Note: The recommended method for creating a new configuration is to copy an existing configuration and modify the copy, as this method minimizes the possibility for error. The following procedure describes the use of this method. However, it is also possible to use standard Siebel Business Applications techniques to create a new record and fill in the necessary fields.

To create a new configuration

1. Navigate to the Administration - Siebel Anywhere screen, then the Upgrade Configurations view.
2. In the Upgrade Configurations list, select an existing configuration record that is similar to the configuration you want to create, and then click the menu button and select Copy Record.

   A highlighted copy of the record that you selected appears.
3. Modify the fields in the copied record to meet your requirements.

   Some fields are described in the following table.

Field Name	Comments
Name	The name of the configuration. **Caution:** Limit the value of the Name field to 91 characters or fewer. Exceeding this limit causes synchronization to fail.
Active	When this check box is selected, it indicates that the configuration is to be version-checked. When you create a new configuration by copying an existing configuration, this check box is cleared by default. Make sure to select it before using the new configuration.
Component Type	This value is either client or server configuration.
Comments	Comments can be entered here.

4. Add appropriate Upgrade Components to the new configuration you just created.

   To do this, see *Adding Components to a Configuration*.

About Assigning Employees to a Configuration

The purpose of assigning employees to configurations is to handle uncommon situations, such as having one particular employee retrieve and test an upgrade kit. In most cases, testing is done using special test accounts and test configurations, and regular employees are associated with a specific configuration based on a ComponentName entry in the client's application CFG file.

This topic is part of *Modifying and Creating Siebel Anywhere Configurations*.

For the procedure for assigning employees to a configuration, see *Assigning Employees to a Configuration*.

For example, for the Siebel Sales application, the default value of ComponentName in the siebel.cfg file would be Siebel Sales Client. This means that everyone using the Siebel Sales application is automatically associated with the Siebel Sales configuration.

Assigning employees to a configuration in the Employees list in the Upgrade Configurations view can be used in place of the entry in the CFG file, or as a method for overriding the entry in the CFG file.

> **CAUTION:** It is strongly recommended that you run *only* the Siebel Web Client (the client type that does not require a Siebel installer) for user accounts that have Siebel administrator responsibilities, to make sure that administrative tasks are performed while connected to the HQ Server, and to make sure that the administrator is not prevented from logging in for reasons related to component versions. If you run the Siebel Developer Web Client for any administrator account, then it is strongly recommended that the account *not* be associated with a Siebel Anywhere configuration. This precaution also helps prevent version-related login problems.

When associating employees to configurations, keep in mind the following recommendations:

- Do not dynamically assign the Siebel administrator to a configuration.
- Do not dynamically assign Siebel Tools users to a configuration.

  Siebel Anywhere *will not* allow an employee to be dynamically associated with more than one configuration. For example, if the employee JSMITH is dynamically associated with Configuration A and the Siebel administrator then associates JSMITH with Configuration B, then JSMITH is automatically disassociated from Configuration A and associated only with Configuration B.

  > **CAUTION:** If an employee is accidentally associated with an incorrect configuration, then that employee is disassociated from the correct configuration and might not have access to upgrades that are distributed to the correct configuration. For this reason, it is best to correct the configuration assignment without delay.

If an employee will use more than one installation of the Siebel client in the same Siebel Business Applications implementation, then do not associate that employee with a configuration. For example, if the employee JSMITH chooses to install and use both Siebel Call Center and Siebel Sales, then do not dynamically associate that employee with a Siebel Anywhere configuration. The reason is the upgrade kit for one configuration will likely be different than the upgrade kit for another configuration.

In this example, the CFG file would be different for the Siebel Call Center and Siebel Sales configurations. If JSMITH were dynamically associated to the Call Center configuration and logged in using the Siebel Sales client, then the version check would only detect that there was a new upgrade kit and would not differentiate between the two installations of the Siebel client.

If the login is specified under an Employees view, then Siebel Anywhere does not check the CFG file any further and only checks the version of the component listed under the associated configuration for that client. Typically, Siebel Anywhere groups users by the product they use, for example Call Center or Sales Client, which is determined by the CFG file the user uses.

> **Note:** The first time a Mobile Web Client connects to the Siebel Remote Server to initialize the local database, Siebel Anywhere does a version check for the client using the Upgrade Configuration specified in the CFG file. Siebel Anywhere cannot detect a dynamic configuration assignment at this time. However, after the Mobile Web Client successfully initializes the local database, Siebel Anywhere can detect any dynamic configuration assignment for that client.

Assigning Employees to a Configuration

This topic describes how to assign employees to a configuration.

For more information about assigning employees to a configuration, see *About Assigning Employees to a Configuration*.

This topic is part of *Modifying and Creating Siebel Anywhere Configurations*.

To assign an employee to a configuration

1. Navigate to the Administration - Siebel Anywhere screen, then the Upgrade Configurations view.
2. In the Upgrade Configurations list, select the configuration to which you want to add an employee, and then click the Employees view tab.
3. In the Employees list, click New.
4. In the Login field of the new record, click the select button.

   From the Pick Employee dialog box, select the employee you want to add, and click OK.
5. Repeat the previous steps to assign additional employees to this configuration.

Removing Employees from a Configuration

Use the following procedure to end an employee's dynamic configuration assignment, which returns that user to the original configuration assignment defined in the user's CFG file.

This topic is part of *Modifying and Creating Siebel Anywhere Configurations*.

To remove an employee from a configuration

1. Navigate to the Administration - Siebel Anywhere screen, then the Upgrade Configurations view.
2. In the Upgrade Configurations list, select the configuration from which you want to remove an employee, and then click the Employees view tab.
3. In the Employees list, select the record for the employee you want to remove from the configuration, and then click the menu button and select Delete Record.

Setting Up Custom Siebel Anywhere Upgrade Components

This topic contains information about creating custom components. An upgrade component is a required infrastructure element for the successful creation and deployment of an upgrade kit.

Siebel Business Applications provide many predefined upgrade components, but you can create additional components if you need them. For example, it would be possible to create an upgrade component to run a batch program in the form of a BAT file.

Siebel Anywhere can upgrade custom components, just as it does the predefined components provided with Siebel Business Applications. For correct operation, however, you must define how Siebel Anywhere will monitor and verify the version number for any custom component.

This task is a step in *Process of Planning and Preparing to Create Upgrade Kits*.

This topic contains the following information:

- *About Monitoring and Verifying Siebel Anywhere Version Numbers*
- *Creating a Custom Component*

About Monitoring and Verifying Siebel Anywhere Version Numbers

Siebel Anywhere uses information from the Locate Method, Locate Information, Version Method, and Version Information settings to determine how it must check the version of a component. Depending on the component, Siebel Anywhere uses some or all of these settings.

This topic is part of *Setting Up Custom Siebel Anywhere Upgrade Components*.

For more information about when a version check occurs for each type of subscriber, see the following table.

The following information describes how the Locate Method and Version Method settings are used, along with the Locate Information and Version Information settings. An example follows.

Locate Method

The purpose of Locate Method is to build a path to a file or other location, such as a registry. This value is then passed to Version Method to read version information. The following table describes some available values for Locate Method, each of which specifies a different way to locate version information. For some values, you must specify additional information for Location Information.

Value	Comments
Siebel Root/Siebel Home	Determines the location based on the client's Siebel root directory. Locate Information is ignored in this case. For example, use this method for Siebel Upgrade Wizard components. Both server and client use Locate Method to retrieve the value of Siebel Root.
Registry	Determines the location based on information in the Microsoft Windows registry. Locate Information would contain the path to read in the registry. For example, use this method for Third Party - Microsoft Word components.
Environment	Determines the location on the client by reading an environment variable. Locate Information would contain the name of the environment variable to read.
CCF	Determines the location on the server from a set of common core facility (CCF) parameters. Locate Information would contain the value from the CCF program context. For example, use this method for a Siebel Server Customer Revisions component.
CFG	Determines the location based on reading the client's active CFG file. Locate Information would contain the section and entry to read from the CFG file, plus an optional value to append to the value read

Value	Comments
	from the CFG file. For example, use this method for a Siebel Client Customer Revisions component or a Siebel Repository Export Import component.

Version Method

This setting provides additional information that is used in conjunction with Locate Method to read the version of a file. The following table describes the available values for Version Method, each of which specifies a different way to determine the version of the component. For some values, you must specify additional information for Version Information.

Value	Comments
File Version	Determines the version based on reading the version of a file. Version Information would contain the name of the file to version check.
Registry	Determines the version based on reading the version from the registry. Version Information would contain the path to read in the registry.
DLL	Determines the version based on executing an entry point in a DLL file. Version Information would contain the name of the DLL file and the name of the entry point. For example, the value *BIN \sslcver.dll,VersionString* executes the function VersionString in the file **`BIN\sslcver.dll`**.
Database	Determines the version based on reading the Siebel database schema. Version Information is ignored in this case.
CFG	Determines the version based on reading the client's active CFG file. Version Information would contain the section and entry to read from the CFG file, plus an optional value to append to the value read from the CFG file. For example, use this method for a Siebel Client Customer Revisions component or a Siebel Repository Export Import component.

Example: How Siebel Anywhere Uses Locate Method and Version Method

The following example shows how Siebel Anywhere checks component versions using the Locate Method and Version Method settings. The following table shows information that appears in the Upgrade Components list for the Siebel Client Customer Revisions__ENU component, along with comments explaining how each setting is used.

Field Name	Value	Comments
Name	Siebel Client Customer Revisions__ENU	The name of the component.
Locate Method	CFG	Information about the location of component version information is stored in the client's active CFG file.

Field Name	Value	Comments
Locate Information	Siebel,ClientRootDir	Information about the location of component version information is stored in the ClientRootDir entry (which is the client installed location) in the [Siebel] section of the client's CFG file. For example, the value of ClientRootDir might be `C:\Siebel\Client` or `C:\Siebel\Client_1`.
Version Method	CFG	The version information is stored in the CFG file specified by the Locate Method and Locate Information settings.
Version Information	Siebel,Version	The version information is stored in the Version entry in the [Siebel] section of the specified CFG file.

Creating a Custom Component

Siebel Anywhere functionality allows you to define your own custom components, if necessary.

This topic is part of *Setting Up Custom Siebel Anywhere Upgrade Components*.

The Siebel administrator *must define* how Siebel Anywhere will monitor and verify the version number for a custom upgrade component. Siebel Anywhere maintains an individual version number for each upgrade component. The version number must be maintained on a Siebel Anywhere subscriber to compare it to the version number in the database. The Siebel administrator can maintain the version number in the application, in a file, or in the Windows registry. For more information, see *About Monitoring and Verifying Siebel Anywhere Version Numbers*.

CAUTION: It is strongly recommended that you use the Copy Record functionality to copy existing upgrade components and simply rename the new upgrade component so that the correct parameters are used for the new upgrade component. This method is described in the following procedure. Make sure that you also update the Locate and Version columns appropriately.

To create a custom component

1. Navigate to the Administration - Siebel Anywhere screen, then the Upgrade Component List view.
2. In the Upgrade Components list, select an existing component that is similar to the component you want to create, and then click the menu button and select Copy Record.

   A highlighted copy of the record that you selected appears.
3. Modify the fields in the copied record to meet your requirements.

   Use a meaningful name to describe the function of the component. Make sure that the Min Version and Max Version fields are blank. This information is filled in when the upgrade kits are applied.

   CAUTION: Limit the value of the Name field to 91 characters or fewer. Exceeding this limit causes synchronization to fail.

4. For any configuration that you will upgrade using your new custom component, you must specify that the component is a related component for the configuration.

   For more information, see *Adding Components to a Configuration*.

Verifying Client Connections to the Siebel File System

The Siebel File System on the server stores the newly created upgrade kits. It is important to verify settings for client connections to the Siebel File System. The connection settings vary depending upon the type of client. In either case these settings are essential to downloading the upgrade kits.

This task is a step in *Process of Planning and Preparing to Create Upgrade Kits*.

Mobile Web Clients connect to the Siebel File System through the Siebel Server. Therefore, set the Siebel File System parameter for the Synchronization Manager server component to the same location as the FileSystem parameter in the CFG file of the user who created the upgrade kit.

Developer Web Clients connect directly to the Siebel File System. Therefore, set the FileSystem parameter in the [ServerDataSrc] section of the client application's CFG file to the appropriate location. Also, the user must have read and write access to that directory. Test for this by having the user open an attachment. If that is successful, then the connection to the File System is set correctly.

For more information about the Siebel File System, see the ***Siebel Installation Guide*** and ***Siebel System Administration Guide***.

Verifying the Availability of Siebel Anywhere

The Siebel Anywhere component group must be available to allow the creation of upgrade kits. The following procedure describes how to verify the availability of this component group.

To check the availability of Siebel Anywhere

1. Navigate to the Administration - Server Management screen, then the Servers view.
2. In the Servers list, select a Server that you want to inspect.
3. Click the Component Groups view tab, and select the Siebel Anywhere component group.
4. Verify that State is set to Online.
5. If State is not Online, then see ***Siebel System Administration Guide*** for information about administering component groups, including instructions for enabling a component group.

Preparing Upgrade Kit Contents

Preparing to create an upgrade kit involves gathering the software, files, or Siebel database schema updates required for the upgrade kit. It can also involve preparing simple batch scripts. The specific process to follow depends on the type of components you are planning to upgrade.

This task is a step in *Process of Planning and Preparing to Create Upgrade Kits*.

> **Note:** Do not use the component type Customer Revision for upgrade kits containing a Siebel runtime repository or CFG file. Instead, use the Siebel Repository ExportImport and Siebel CFG File component types, respectively, for these upgrades. Using the proper types will allow proper version checking.

The following topics describe specific steps to take for certain upgrade kit types:

- *Preparing Contents for a Siebel Upgrade Wizard Upgrade Kit*
- *Preparing Contents for a Siebel Client Executables Upgrade Kit*
- *About Reducing the Size of Siebel Client Executables Upgrade Kits*
- *Preparing Contents for a Siebel Patchset Upgrade Kit*
- *Preparing Contents for Runtime Repository or Third Party Software Upgrade Kits*

Preparing Contents for a Siebel Upgrade Wizard Upgrade Kit

You must create a Siebel Upgrade Wizard upgrade kit whenever there are changes to the Upgrade Wizard in a Siebel CRM release. For more information about whether a specific release contains such changes, see the release-specific documentation for the release.

This topic is part of *Preparing Upgrade Kit Contents*.

A Siebel Upgrade Wizard upgrade kit must contain a set of DLL files, along with the Upgrade Wizard executable file, siebupg.exe. The following procedure describes how to prepare the contents for such an upgrade kit.

To prepare the contents for a Siebel Upgrade Wizard upgrade kit

1. Create a new directory to hold the Upgrade Wizard executable program and related DLL files.
   Make a note of the directory name, because you must specify it while defining the upgrade kit.
2. If you have not already done so, then install the client software for the new Siebel CRM release on a test computer or a Siebel administrator's computer. Make a note of where you installed it.
3. Locate the client `BIN` directory within the installation directory of the new Siebel CRM release.
   The installation location might resemble the following:
   ```
   C:\Siebel\Client\BIN
   ```
4. Copy the following files from the release client `BIN` directory into the new directory that you created previously.
   - siebupg.exe
   - MSVCR71D.dll
   - MFC71UD.dll
   - MSVCP71D.dll
   - libarm.dll
   - pw32a.dll
   - srlcver.dll
   - sslcacln.dll
   - sslccore.dll

- sslcdb.dll
- sslcevt.dll
- sslclm.dll
- sslcnapi.dll
- sslcns.dll
- sslcos.dll
- sslcosa.dll
- sslcosd.dll
- sslcosslimpl.dll
- sslcrsa.dll
- sslcrsaimpl.dll
- sslcrsa56.dll
- sslcsc.dll
- sslcscr.dll
- sslcsecm.dll
- sslcshar.dll
- sslcsnns.dll
- sslcsnsc.dll
- sslcsrd.dll
- sslcsrvr.dll
- sslcsym.dll
- sslcupg.dll
- sslcver.dll

Preparing Contents for a Siebel Client Executables Upgrade Kit

You can use the Siebel Client Executables components to deliver Siebel Innovation Pack releases and Siebel language packs to Siebel Mobile or Developer Web Clients.

This topic is part of *Preparing Upgrade Kit Contents*.

Note: For Siebel Patchset releases, see *Preparing Contents for a Siebel Patchset Upgrade Kit*.

Siebel Innovation Pack releases consist of a set of language-independent files (called *base* files) and one or more sets of language-specific files. The following components are available:

- Upgrade kits using the Siebel Client Executables component are used to distribute base files. Language-specific files can also be included in this upgrade kit, along with the base files. You can use this component for standard installation or delta installation upgrade kits.

- Upgrade kits using the Siebel Client Executables__*language-code* components are optionally used to distribute language-specific files. You can use these components only for delta installation upgrade kits.

For more information about delta installation upgrade kits, see *About Reducing the Size of Siebel Client Executables Upgrade Kits* and *Roadmap for Creating a Delta Installation Siebel Client Executables Upgrade Kit*.

The following procedures describe how to prepare files for inclusion in base client executables upgrade kits.

To prepare files for inclusion in Siebel Client Executables upgrade kits (for standard installation upgrade kits)

1. Create a new, network-accessible UNC directory to use for storing the base client executable files in the Siebel Innovation Pack release.

   An example name for this directory might be `\\` computer_name `\siebel\base`.
2. Navigate to the parent directory, within the network image file hierarchy, of the Siebel client directory for the Siebel Innovation Pack release. Navigate to a directory like the following:

   `Windows\Client`
3. Copy the `Siebel_Web_Client` directory to the clipboard.
4. Paste the directory that you copied to the directory that you created in the previous step.

Related Topics

About Reducing the Size of Siebel Client Executables Upgrade Kits

Roadmap for Distributing a Siebel Innovation Pack or Patchset Release

Roadmap for Creating a Delta Installation Siebel Client Executables Upgrade Kit

Defining a Siebel Client Executables Upgrade Kit

Controlling the Order of Upgrade Kit Installation

Distributing Additional Language Packs

About Reducing the Size of Siebel Client Executables Upgrade Kits

This topic describes a feature called the delta installation upgrade kit type for a Siebel Client Executables upgrade kit.

This topic is part of *Preparing Upgrade Kit Contents*.

It is possible to reduce the size of some Siebel Client Executables upgrade kits, which also reduces the time it takes users to retrieve and install these upgrade kits. This reduction in upgrade kit size is accomplished by creating the *delta installation* type of Siebel Client Executables upgrade kit (as opposed to a standard installation upgrade kit). Delta installation upgrade kits contain condensed information about specific *changes* to client executable files, instead of an actual set of the new executable files.

When you create a Siebel Client Executables upgrade kit, you choose whether to create it as a delta installation upgrade kit or a standard installation upgrade kit.

Note: Delta installation upgrade kits are suitable for delivering Siebel Innovation Pack releases that contain relatively few changes to client executable files. For example, you could use a delta installation upgrade kit to upgrade Siebel CRM client executables from Siebel CRM version 17.0 to version 17.1. Delta installation upgrade kits are *not* suitable for major upgrades, such as to upgrade Siebel CRM from version 8.1.1 or version 8.2.2 to Siebel Innovation Pack 2017 (Siebel CRM version 17.0). In addition, delta installation upgrade kits are not suitable for adding languages.

You can create delta installation upgrade kits for the Siebel Client Executables base (language-independent) component or for each language-specific Siebel Client Executables__*language-code* component that is for a language that your users already have installed.

CAUTION: If you want to add an additional language for your users, then do not use a delta installation upgrade kit to add that language. Instead, use a standard installation upgrade kit. That is, create an upgrade kit for the Siebel Client Executables__*language-code* component and select Standard Install as the installation type for the upgrade kit. Delta installation upgrade kits are not supported for the addition of a new language. If a delta installation upgrade kit is incorrectly used to install a new language, then it can damage some client executable files.

Before defining a delta installation upgrade kit, a Siebel administrator must use the Siebel Patch Utility (siebpatch.exe) to create a *delta patch file*. This is a file that contains information about the changes that were made to client executable programs between two specific Siebel releases. For details, see *Creating a Delta Patch File*.

Users retrieve and install delta installation upgrade kits just as if the upgrade kits contained complete sets of client executable files. When a delta installation upgrade kit is installed, the Upgrade Wizard applies the changes that are described in the upgrade kit to the user's existing client executable files, rather than replacing the user's files.

Depending on the release you are installing, users might need to upgrade the Upgrade Wizard before installing a delta installation upgrade kit. In this case, a Siebel administrator must define an Upgrade Wizard upgrade kit, and make any delta installation upgrade kits dependent on that Upgrade Wizard kit. For more information about whether this requirement applies to you, see the release-specific documentation for the release you want to install.

For more information about the process of creating delta installation upgrade kits, see *Roadmap for Creating a Delta Installation Siebel Client Executables Upgrade Kit*.

Preparing Contents for a Siebel Patchset Upgrade Kit

You can use the Siebel Quick Fix component to deliver Siebel Patchset releases to Siebel Mobile or Developer Web Clients.

This topic is part of *Preparing Upgrade Kit Contents*.

Note: For Siebel Innovation Pack releases, see *Preparing Contents for a Siebel Client Executables Upgrade Kit*.

The following procedure describes how to prepare files for inclusion in Siebel Patchset upgrade kits.

To prepare files for inclusion in Siebel Patchset upgrade kits

1. Create a new, network-accessible UNC directory to use for storing the files in the Siebel Patchset release.

   An example name for this directory might be \\ computer_name `\siebel\patchset`.

2. Navigate to the parent directory, within the network image file hierarchy, of the Siebel client directory for the Siebel Patchset release. Navigate to a directory like the following:

   `Windows\Client`
3. Copy the `Siebel_Web_Client` directory to the clipboard.
4. Paste the directory that you copied to the directory that you created in the previous step.

Preparing Contents for Runtime Repository or Third Party Software Upgrade Kits

To prepare for creating an upgrade kit for a Siebel runtime repository update, third-party application, or other non-Siebel file, copy the files to a directory on the Siebel Server from which you will run the Upgrade Kit Builder server component, or to a directory that is accessible from that server.

This topic is part of *Preparing Upgrade Kit Contents*.

Siebel Anywhere can install third-party applications or other files using one of the following methods:

- Executing a batch script or installation program.
- Copying files to a specified location.

> **Note:** Before deploying an upgrade kit, make sure that any files, scripts, or other programs required to install third-party applications contain the necessary data and are fully tested.

Creating a Delta Patch File

The following procedure describes how to create a delta patch file for use in a delta installation type of Siebel Client Executables upgrade kit. A delta patch file contains information about how client executables have changed between two specific versions of Siebel CRM.

This task is a step in *Roadmap for Creating a Delta Installation Siebel Client Executables Upgrade Kit*.

For example, you could create a delta patch file that contains information about how Siebel CRM client executables changed between version 8.1.1.7 or version 8.2.2.2 and version 17.0. In this example, to upgrade Siebel CRM version 8.1.1.7 or version 8.2.2.2 client executables to version 17.0, you could use a delta installation type of Siebel Client Executables upgrade kit that contains that delta patch file.

For general information about delta installation upgrade kits, see *About Reducing the Size of Siebel Client Executables Upgrade Kits*.

To create a delta patch file for use in a delta installation Siebel Client Executables upgrade kit

1. On a computer that has client software for the new release already installed, use a text editor to inspect the siebdp.rtc file.

   This file is typically located in the `BIN` subdirectory for the client software. For example, if the client software was originally installed in `c:\Siebel\Client`, then the siebdp.rtc file would be located in `c:\Siebel\Client\BIN`.
2. If your Siebel Business Applications implementation contains files or directories that you do not want inspected for changes when the delta patch file is created, then add those files or directories to the siebdp.rtc file, and then save the file. The correct location for your additions is after the files and directories that are already listed in the following section of the siebdp.rtc file:

```
IGNORE *.log, *.err, *.cfg, *.file
IGNORE *_setup.reg
IGNORE Uninst*.bat
IGNORE upgrade*.txt
IGNORE siebupg2.exe
IGNORE siebupg1.exe
IGNORE *.spf
IGNORE *.tmp
IGNORE bin\ssceolrs.dll
IGNORE bin\ssupgpures.dll
IGNORE bin\siebdp.rtc
IGNORE bin\siebdp.rtct
IGNORE bin\siebdpnew.rtd
IGNORE install.platform
IGNORE lan.rsp
IGNORE lan.rsp.sbs
IGNORE lan.rsp.sbs
IGNORE en.rsp
IGNOREDIR _uninst
IGNOREDIR UPGRADE
IGNOREDIR packager
IGNOREDIR temp
IGNOREDIR local
IGNOREDIR log
IGNOREDIR cfgtoollogs
IGNOREDIR CLASSES
IGNOREDIR diagnostics
IGNOREDIR install
IGNOREDIR inventory
IGNOREDIR jdk
IGNOREDIR jr
IGNOREDIR OPatch
IGNOREDIR oui
IGNOREDIR XMLP
; You may customize the following section to include/exclude folder and files.
; To ignore file, use: IGNORE $filename
; To ignore directory, use: IGNOREDIR $directoryname
```

> **CAUTION:** Do not change the entries in the earlier part of the siebdp.rtc file. Changing the earlier entries can prevent the correct creation of the delta patch file.

   If you want to add a comment to the file, then place the comment on its own line, and use a semi-colon (;) as the first character of the line.

3. Open a Command Prompt window and navigate to the `BIN` directory for the installed client software (which includes siebdp.rtc and siebpatch.exe), such as the following:

   ```
   C:\Siebel\Client\BIN
   ```

4. In the Command Prompt window, enter a command like the following, substituting the appropriate path to the directory that contains the siebdp.rtc file and substituting your applicable language (one of the languages installed on the Siebel Mobile Web Client):

   ```
   siebpatch /l fra /c C:\Siebel\Client\BIN\siebdp.rtc
   ```

   The Siebel Patch Utility window appears.

5. Use the information in the following table to respond to the prompts in the Siebel Patch Utility, and then click Next to start the creation of the delta patch file and display progress indicators.

Siebel Patch Utility Field or Prompt	Comments
Location of the current Siebel client	Enter the path to the directory where the older version of the Siebel client software is installed (the version that your users have now).
Location of the new Siebel client	Enter the path to the directory where the newer version of Siebel client software is installed (the version to which your users will upgrade).
Language	If you are creating the delta patch file for the base (language-independent) portion of a Siebel client installation, for a Siebel Client Executables upgrade kit, then select BASE. If you are creating the delta patch file for the base portion, plus all language-specific portions, of a Siebel client installation, for a Siebel Client Executables upgrade kit, then leave this field blank. If you are creating the delta patch file for a language-specific portion of a Siebel client installation, for a Siebel Client Executables__*language-code* upgrade kit, then select the appropriate language code.
Output File	Specify a path and file name for the delta patch file you are creating. (This is the file that will be included in the upgrade kit.) Use RTP as the file name extension for this file. If you want, you can click the Browse button to navigate to the directory where you want the file to be placed. **Note:** If you specify *filename*.rtp as the name for the delta patch file, then both *filename*.rtp and *filename*.rtd are automatically created. The RTD file is a text file that provides information about the files that have changed between the two specified versions of the software, along with information about any compression of those files that occurred when *filename*.rtp was created.

6. When the Siebel Patch Utility displays a message that the output package has been generated successfully, click Finish to exit from the utility.

7. As needed, repeat this entire procedure to create a separate delta patch file for use in each of the following upgrade kit types:
    - Delta installation Siebel Client Executables (language-independent base upgrade kit)
    - (Optional) Delta installation Siebel Client Executables__*language-code* (for each language that is used in your Siebel Business Applications implementation)

Synchronizing Server Components

Before you create your first upgrade kit, you must synchronize server components. If synchronization has not been done, then attempts to define upgrade kits fail. For more information about synchronizing server components, see *Siebel System Administration Guide* .

To synchronize server components before defining upgrade kits

1. Navigate to the Administration - Server Configuration screen, then Enterprises, then the Component Groups view.
2. In the Component Groups list, verify that the Siebel Anywhere component group is enabled.
3. Click the Component Definitions tab.
4. In the Component Definitions list, use standard query techniques to locate the record for the Siebel Anywhere component group, and click Enable if it is not currently active.
5. Click Synchronize.
6. Restart the Siebel Server where the Siebel Anywhere components are enabled.

4 Defining Upgrade Kits

Defining Upgrade Kits

This chapter provides information about and general instructions for running the Upgrade Kit Wizard, followed by specific instructions for defining several types of upgrade kits, and instructions for viewing upgrade kit properties, after the upgrade kits have been defined.

It includes the following topics:

- *About Defining Upgrade Kits*
- *Running the Upgrade Kit Wizard*
- *Defining a Siebel Upgrade Wizard Upgrade Kit*
- *Defining a Siebel Configuration File (CFG) Upgrade Kit*
- *Defining a Siebel Database Schema Upgrade Kit*
- *Defining a Siebel Client Executables Upgrade Kit*
- *Roadmap for Distributing a Siebel Innovation Pack or Patchset Release*
- *Roadmap for Creating a Delta Installation Siebel Client Executables Upgrade Kit*
- *Defining a Siebel Patchset Upgrade Kit*
- *Defining a Siebel Repository Export Import Upgrade Kit*
- *Defining a Third Party Software Upgrade Kit*
- *Defining a Siebel Workspace Seed Import*
- *Defining a Siebel Client Customer Revisions Upgrade Kit*
- *Defining a Custom Component Upgrade Kit*
- *Viewing Upgrade Kit Properties*

About Defining Upgrade Kits

Defining an upgrade kit is the process of running the Upgrade Kit Wizard and entering data concerning a single Siebel Anywhere component that is to be upgraded. After all necessary data has been provided, the Upgrade Kit Wizard invokes the Upgrade Kit Builder server component, which uses the gathered data to build the upgrade kit.

> **Note:** It is important to gather the data to be entered before you run the Upgrade Kit Wizard. For more information about planning your upgrade kit and gathering the necessary data, see *Planning and Preparing to Use Siebel Anywhere* .

For more information about running the Upgrade Kit Wizard, see *Running the Upgrade Kit Wizard*. For general information about the Upgrade Kit Wizard and the Upgrade Kit Builder, see *Siebel Anywhere Wizards and Utilities*.

For language-specific upgrade kits, use the instructions for the appropriate upgrade kit type. For example, use the instructions for defining a CFG upgrade kit to define a language-specific CFG upgrade kit.

Note: If your upgrade includes multiple components, then use the instructions in this chapter to create an upgrade kit for each component.

After you define your upgrade kit, you must proceed to activate, apply, and distribute it. For more information about these tasks, see *Activating, Applying, and Distributing Upgrade Kits*.

Running the Upgrade Kit Wizard

All upgrade kits must be defined using the Upgrade Kit Wizard. This topic provides general instructions for running the Upgrade Kit Wizard. For background information about the Upgrade Kit Wizard, see *Siebel Anywhere Wizards and Utilities*. See also *About Defining Upgrade Kits*.

Note: The following procedure describes how to start the Upgrade Kit Wizard for any type of upgrade kit. Subsequent steps and information required vary depending on the type of upgrade kit.

To start the Upgrade Kit Wizard

1. Navigate to the Administration - Siebel Anywhere screen, then the Upgrade Kits view.

   The Upgrade Kits view appears.
2. Click Auto Create to start the Upgrade Kit Wizard.

   The Upgrade Kit Wizard - Select Upgrade Component page appears.
3. Follow the prompts to specify all parameters necessary for the type of upgrade kit you are creating.

Related Topics

Defining a Siebel Upgrade Wizard Upgrade Kit

Defining a Siebel Configuration File (CFG) Upgrade Kit

Defining a Siebel Database Schema Upgrade Kit

Defining a Siebel Client Executables Upgrade Kit

Defining a Siebel Patchset Upgrade Kit

Defining a Siebel Repository Export Import Upgrade Kit

Defining a Third Party Software Upgrade Kit

Defining a Siebel Workspace Seed Import

Defining a Siebel Client Customer Revisions Upgrade Kit

Defining a Custom Component Upgrade Kit

Defining a Siebel Upgrade Wizard Upgrade Kit

This topic provides specific instructions for defining a Siebel Upgrade Wizard upgrade kit using the Upgrade Kit Wizard. Upgrade kits of this type are used to upgrade the Upgrade Wizard as part of a Siebel CRM release.

CAUTION: When a Siebel CRM release includes an update to the Upgrade Wizard, upgrade tasks must be performed in the following order to accomplish upgrades correctly. First, install any necessary Siebel CRM software for the Siebel Gateway and Siebel Server, for Siebel Innovation Pack 2017. Otherwise, version numbering for subsequent upgrade tasks might be incorrect. After performing Siebel Server installations, complete the upgrade to the Upgrade Wizard for each client before you upgrade any other software components for the client. This allows the latest version of the Upgrade Wizard to be used to accomplish the upgrades of other components.

To define a Siebel Upgrade Wizard upgrade kit

1. Gather the information and files you will need, as described in *Planning and Preparing to Use Siebel Anywhere*.
2. Start the Upgrade Kit Wizard, as described in *Running the Upgrade Kit Wizard*.
3. Use the information you have gathered in the previous step to respond to the prompts in the Upgrade Kit Wizard.

   Note: The Minimum Old Version and Maximum Old Version settings are automatically set to NULL for history-independent component types, including Siebel Upgrade Wizard, indicating that there are no prerequisite versions required for using the upgrade kit. The value for New Version is also automatically supplied for Siebel Upgrade Wizard upgrade kits.

4. When you have finished specifying data about the upgrade kit you are defining, click Finish to pass the request to the Upgrade Kit Builder server component and to exit from the Upgrade Kit Wizard.

   A new row for the new upgrade kit appears in the Upgrade Kits list, with Status set to Request Submitted. For more information about the upgrade kit information available in this list and other lists, see *Viewing Upgrade Kit Properties*.

   After you define your upgrade kit, you must proceed to activate, apply, and distribute it. For more information about these tasks, see *Activating, Applying, and Distributing Upgrade Kits* .

The following information is presented in approximately the sequence used by the Upgrade Kit Wizard.

Wizard Element	Element Type	Comments
Upgrade Component	Drop-down list	The name of the component the upgrade kit will install or upgrade. Select Siebel Upgrade Wizard.
UNC Path for File Folder	Text field	The Universal Naming Convention (UNC) path to the directory that contains the siebupg.exe file and its accompanying DLL files for the upgrade. Identify a network-accessible location.

Wizard Element	Element Type	Comments
New Version	Text field	The version number that the component being upgraded will have after the upgrade kit is installed. The default value is 1 greater than the current version for the component to be upgraded. For more information about choosing version values, see *Determining Version-Setting Values*.
Activate Upgrade Kit	Check box	When this check box is selected, the files to be included in the upgrade kit will be compressed into a single archive on the Siebel File System automatically. You can also perform activation manually, as described in *Activating, Applying, and Distributing Upgrade Kits* . For most upgrade kits, it is recommended that you select this check box. However, if you are creating an upgrade kit that is dependent on another upgrade kit, then clear this check box. For more information about working with dependent upgrade kits, see *Controlling the Order of Upgrade Kit Installation*.
Upgrade Kit Title	Text field	An identifier for the upgrade kit. The default value is the value of Upgrade Component Name followed by a space and the value of New Version. However, you can modify the title while defining the upgrade kit. After the upgrade kit is defined, the title appears in the Name field of the Upgrade Kits list.
Description	Text field	This field is available for comments about the upgrade kit. Comments entered here are displayed in the Upgrade Kits list.

Defining a Siebel Configuration File (CFG) Upgrade Kit

This topic provides specific instructions for defining a Siebel Configuration File upgrade kit using the Upgrade Kit Wizard. Upgrade kits of this type are used to upgrade Siebel CFG files, and are always language-specific.

To define a Siebel Configuration File upgrade kit

1. Gather the information and files you will need, as described in *Planning and Preparing to Use Siebel Anywhere*.
2. Start the Upgrade Kit Wizard, as described in *Running the Upgrade Kit Wizard*.
3. Use the information you have gathered in the previous step to respond to the prompts in the Upgrade Kit Wizard.

4. When you have finished specifying data about the upgrade kit you are defining, click Finish to pass the request to the Upgrade Kit Builder server component and to exit from the Upgrade Kit Wizard.

   A new row for the new upgrade kit appears in the Upgrade Kits list, with Status set to Request Submitted. For more information about the upgrade kit information available in this list and other lists, see *Viewing Upgrade Kit Properties*.

   After you define your upgrade kit, you must proceed to activate, apply, and distribute it. For more information about these tasks, see *Activating, Applying, and Distributing Upgrade Kits*.

The following information is presented in approximately the sequence used by the Upgrade Kit Wizard.

Wizard Element	Element Type	Comments
Upgrade Component	Drop-down list	The name of the component the upgrade kit will install or upgrade. Select an option like Siebel Sales CFG__ENU or Siebel Call Center CFG__FRA.
Files to Add	Text field	The name of the CFG file to include in the upgrade kit. This field is populated either by entering a file name and path, or by clicking Browse and choosing a file from the directory listings displayed.
Browse	Button	This button displays a standard dialog box for browsing and choosing a file.
Minimum Old Version	Text field	The earliest component version that can download and install the upgrade kit. When this field is blank, this indicates that there are no prerequisite versions required for using the upgrade kit. The default value is the same as the current component version recorded in the database. For more information about choosing version values, see *Determining Version-Setting Values*.
Maximum Old Version	Text field	The latest component version that can download and install the upgrade kit. When this field is blank, this indicates that there are no prerequisite versions required for using the upgrade kit. For more information about choosing version values, see *Determining Version-Setting Values*.
New Version	Text field	The version number that the component being upgraded will have after the upgrade kit is installed. The default value is 1 greater than the current version for the component to be upgraded. For more information about choosing version values, see *Determining Version-Setting Values*.
Activate Upgrade Kit	Check box	When this check box is selected, the files to be included in the upgrade kit will be compressed into a single archive on the Siebel File System

Wizard Element	Element Type	Comments
		automatically. You can also perform activation manually, as described in *Activating, Applying, and Distributing Upgrade Kits* For most upgrade kits, it is recommended that you select this check box. However, if you are creating an upgrade kit that is dependent on another upgrade kit, then clear this check box. For more information about working with dependent upgrade kits, see *Controlling the Order of Upgrade Kit Installation*.
Apply Versions	Check box	This check box is displayed only if you previously selected the Activate Upgrade Kit check box in the Upgrade Kit Wizard. When this check box is selected, the compiled information string in the database will be updated automatically with the component version information for this upgrade kit. You can also apply versions manually, as described in *Activating, Applying, and Distributing Upgrade Kits* For most upgrade kits, it is recommended that you select this check box. However, if you are creating an upgrade kit that is dependent on another upgrade kit, then clear this check box. For more information about working with dependent upgrade kits, see *Controlling the Order of Upgrade Kit Installation*.
Required Upgrade Kit	Check box	This check box is displayed only if you previously selected the Apply Versions check box in the Upgrade Kit Wizard. When this check box is selected, the upgrade kit will be required regardless of the previous versions installed. (Min Version and Max Version are set equal to the value of New Version, but this change is not visible in the Upgrade Kit Wizard.)
Upgrade Kit Title	Text field	An identifier for the upgrade kit. The default value is the value of Upgrade Component Name followed by a space and the value of New Version. However, you can modify the title while defining the upgrade kit. After the upgrade kit is defined, the title appears in the Name field of the Upgrade Kits list.
Comments	Text field	This field is available for comments about the upgrade kit. Comments entered here are displayed in the Upgrade Kits list.

Defining a Siebel Database Schema Upgrade Kit

This topic provides specific instructions for defining a Siebel Database Schema upgrade kit using the Upgrade Kit Wizard.

Installing a Siebel Database Schema upgrade kit synchronizes the logical and physical schemas on the Mobile Web Client and regional databases.

> **CAUTION:** In an IBM DB2 environment, before creating Siebel Database Schema upgrade kits, you must drop all customized views and triggers. Otherwise, the upgrade kit will fail.

To define a Siebel Database Schema upgrade kit

1. Gather the information you will need, as described in *Planning and Preparing to Use Siebel Anywhere* .
2. Start the Upgrade Kit Wizard, as described in *Running the Upgrade Kit Wizard*.
3. Use the information you have gathered in the previous step to respond to the prompts in the Upgrade Kit Wizard.
4. When you have finished specifying data about the upgrade kit you are defining, click Finish to pass the request to the Upgrade Kit Builder server component and to exit from the Upgrade Kit Wizard.

   A new row for the new upgrade kit appears in the Upgrade Kits list, with Status set to Request Submitted. For more information about the upgrade kit information available in this list and other lists, see *Viewing Upgrade Kit Properties*.

   After you define your upgrade kit, you must proceed to activate, apply, and distribute it. For more information about these tasks, see *Activating, Applying, and Distributing Upgrade Kits*

   > **Note:** Make sure to check the log file after Upgrade Kit Builder creates a Siebel Database Schema upgrade kit. Make sure that error messages or warnings, if any, are nonfatal.

   .

The following information is presented in approximately the sequence used by the Upgrade Kit Wizard.

Wizard Element	Element Type	Comments
Upgrade Component	Drop-down list	The name of the component the upgrade kit will install or upgrade. Select Siebel Database Schema.
ODBC Data Source	Text field	The name of the ODBC data source used to connect to the HQ database.
User Name	Text field	The Siebel administrator login ID used to connect to the database.

Wizard Element	Element Type	Comments
User Password	Text field	The Siebel administrator password used to connect to the database.
Schema Qualifier	Text field	(IBM DB2 for z/OS only) the name used to qualify all database objects created that are required by Siebel Business Applications.
Privileged User ID	Text field	Required. The user account that has the necessary database authority and privileges to perform operations required to implement Siebel Business Applications, including creating, accessing, and modifying Siebel database objects and native database objects. These environments have rigid controls on user identification: accounts must correspond to a real person. For a database other than IBM DB2 for z/OS, this account is the same account as table owner.
Privileged User Password	Text field	The Privileged User's password on the Regional Database. For a database other than IBM DB2 for z/OS, this password is the table owner's password.
Table Space	Text field	(Oracle Database or IBM DB2 only) The database tablespace or segment for Siebel tables. Tablespaces are created during initial installation or by the DBA. Obtain this value from your DBA or through a SQL query on the server database.
Index Space	Text field	(Oracle Database or IBM DB2 only) The database tablespace or segment for Siebel indexes. Obtain this value from your DBA through a SQL query on the server database.
16K Table Space	Text field	(IBM DB2 only) Obtain the optional parameter from your DBA or through a SQL query on the server database.
32K Table Space	Text field	(IBM DB2 only) Obtain the optional parameter from your DBA or through a SQL query on the server database.
Table Group File	Text field	(IBM DB2 for z/OS only) Obtain the optional parameter from your DBA or through a SQL query on the server database.
Minimum Old Version	Read-only text field	The earliest component version that can download and install the upgrade kit. This setting is automatically set to NULL for history-independent component types, including Siebel Database Schema, indicating that there are no prerequisite

Wizard Element	Element Type	Comments
		versions required for using the upgrade kit. For more information about choosing version values, see *Determining Version-Setting Values*.
Maximum Old Version	Read-only text field	The latest component version that can download and install the upgrade kit. This setting is automatically set to NULL for history-independent component types, including Siebel Database Schema, indicating that there are no prerequisite versions required for using the upgrade kit. For more information about choosing version values, see *Determining Version-Setting Values*.
New Version	Text field	The version number that the component being upgraded will have after the upgrade kit is installed. The default value is 1 greater than the current version for the component to be upgraded. For more information about choosing version values, see *Determining Version-Setting Values*.
Activate Upgrade Kit	Check box	When this check box is selected, the data to be included in the upgrade kit will be compressed into a single archive on the Siebel File System automatically. You can also perform activation manually, as described in *Activating, Applying, and Distributing Upgrade Kits* . For most upgrade kits, it is recommended that you select this check box. However, if you are creating an upgrade kit that is dependent on another upgrade kit, then clear this check box. For more information about working with dependent upgrade kits, see *Controlling the Order of Upgrade Kit Installation*.
Apply Versions	Check box	This check box is displayed only if you previously selected the Activate Upgrade Kit check box in the Upgrade Kit Wizard. When this check box is selected, the compiled information string in the database will be updated automatically with the component version information for this upgrade kit. You can also apply versions manually, as described in *Activating, Applying, and Distributing Upgrade Kits* . For most upgrade kits, it is recommended that you select this check box. However, if you are creating an upgrade kit that is dependent on another upgrade kit, then clear this check box. For more information about working with dependent upgrade kits, see *Controlling the Order of Upgrade Kit Installation*.

Wizard Element	Element Type	Comments
Required Upgrade Kit	Check box	This check box is displayed only if you previously selected the Apply Versions check box in the Upgrade Kit Wizard. When this check box is selected, the upgrade kit will be required regardless of the previous versions installed. (Min Version and Max Version are set to the value of New Version, but this change is not visible in the Upgrade Kit Wizard.) Make Siebel Database Schema upgrade kits required when they are distributed to users in a production environment. However, it is recommended that you create and test the upgrade kit as an optional upgrade kit before making it required. For more information about changing an optional upgrade kit to be required, see *Converting an Optional Upgrade Kit to a Required Upgrade Kit*.
Upgrade Kit Title	Text field	An identifier for the upgrade kit. The default value is the value of Upgrade Component Name followed by a space and the value of New Version. However, you can modify the title while defining the upgrade kit. After the upgrade kit is defined, this identifier appears in the Name field of the Upgrade Kits list.
Comments	Text field	This field is available for comments about the upgrade kit. Comments entered here are displayed in the Upgrade Kits list.

Note: To avoid unnecessary download operations, it is recommended that you keep only one upgrade kit for history-independent components, including Siebel Database Schema upgrade kits. Depending on your preference, you can either delete or deactivate previous upgrade kits for a history-independent component. Deleting an upgrade kit increases available space in the Siebel File System. Deactivating an upgrade kit prevents use of the upgrade kit, while keeping it available in case you need it unexpectedly.

For detailed information about using a Siebel Database Schema upgrade kit, see *Process of Updating the Siebel Database Schema*.

Defining a Siebel Client Executables Upgrade Kit

This topic provides specific instructions for defining an upgrade kit for a Siebel Client Executables component using the Upgrade Kit Wizard. You can deliver a Siebel Innovation Pack release with these components.

Instructions in this topic apply to the following two component types:

- **Siebel Client Executables.** Used for language-independent portions of Siebel Innovation Pack releases, known as *base* portions. Language-specific files can also be included in this upgrade kit, along with the base files. You can use this component for standard installation or delta installation upgrade kits.
- **Siebel Client Executables__** *language-code* **.** Optionally used for language-specific portions of Siebel Innovation Pack releases. You can use these components only for delta installation upgrade kits.

Instructions apply to both the standard installation and the delta installation types of Siebel Client Executables upgrade kit.

This task is a step in *Roadmap for Creating a Delta Installation Siebel Client Executables Upgrade Kit*.

For more information about delta installation upgrade kits, see *About Reducing the Size of Siebel Client Executables Upgrade Kits*.

> **Note:** For Siebel Patchset releases, use Siebel Patchset upgrade kits, as described in *Defining a Siebel Patchset Upgrade Kit*.

The following procedure describes how to define either a Siebel Client Executables upgrade kit for the base component or a Siebel Client Executables__*language-code* upgrade kit for a language-specific component.

To define a Siebel Client Executables upgrade kit (base or language-specific)

1. Gather the information and files you will need, as described in *Planning and Preparing to Use Siebel Anywhere*.
2. Start the Upgrade Kit Wizard, as described in *Running the Upgrade Kit Wizard*.
3. Use the information you have gathered in the previous steps to respond to the prompts in the Upgrade Kit Wizard.

   > **Note:** The Minimum Old Version and Maximum Old Version settings are automatically set to NULL for history-independent component types, including Siebel Client Executables, indicating that there are no prerequisite versions required for using the upgrade kit. The value for New Version is also automatically supplied for Siebel Client Executables upgrade kits. These settings are displayed only in the final screen of the Upgrade Kit Wizard.

4. When you have finished specifying data about the upgrade kit you are defining, click Finish to pass the request to the Upgrade Kit Builder server component and to exit from the Upgrade Kit Wizard.
   A new row for the new upgrade kit appears in the Upgrade Kits list, with Status set to Request Submitted. For more information about the upgrade kit information available in this list and other lists, see *Viewing Upgrade Kit Properties*.
   After you define your upgrade kit, you must proceed to activate, apply, and distribute it. For more information about these tasks, see *Planning and Preparing to Use Siebel Anywhere*.

The following information is presented in approximately the sequence used by the Upgrade Kit Wizard.

Wizard Element	Element Type	Comments
Upgrade Component	Drop-down list	The name of the component the upgrade kit will install or upgrade. Select Siebel Workspace Seed Import.

Wizard Element	Element Type	Comments
Select Type of Installation	Radio button	Determines the size and nature of the upgrade kit. If you want the upgrade kit to contain a complete set of the applicable client executable files, then select Standard Install. If you want the upgrade kit to contain condensed information about *only* the executable files that have changed in a particular Siebel Innovation Pack release, then select Delta Install. For more information about delta installation upgrade kits, see *About Reducing the Size of Siebel Client Executables Upgrade Kits* and *Roadmap for Creating a Delta Installation Siebel Client Executables Upgrade Kit*.
UNC Path for Master Installation	Text field	This field is displayed only if you previously selected Standard Install in the Upgrade Kit Wizard. The Universal Naming Convention (UNC) path to the directory that contains the files that you prepared for the upgrade, as described in *Preparing Contents for a Siebel Client Executables Upgrade Kit*. Identify a network-accessible location, as follows: An example path is: \\ computer_name **\siebel\base**
UNC Path for Siebel Delta Patch File	Text field	This field is displayed only if you previously selected Delta Install in the Upgrade Kit Wizard. The Universal Naming Convention (UNC) path to the RTP delta patch file that was created by using the Siebel Patch Utility (siebpatch.exe). Specify the complete directory path *and* file name.
The path of startup program	Text field	Specifies the location of the file to be invoked to install the upgrade kit, within the directory specified with UNC Path for Master Installation. Use the default value.
Uninstall previous Siebel version	Check box	Do not select this check box. **Caution:** When this check box is selected, the Upgrade Wizard automatically uninstalls *all* existing Siebel applications before proceeding with installation of the upgrade. **Note:** Siebel Innovation Pack releases are designed to be applied, without the need for previous patches to be uninstalled. However, occasionally it is useful for testing or troubleshooting to uninstall the current release. For more information about uninstalling a release *without* uninstalling the prior installed release, where applicable, see *Uninstalling a Delta Patch*.
Enter command line arguments	Text field	Specifies any command-line arguments to be used by the Siebel Innovation Pack client installer executable program. Use the default value. In general, change the default command-line arguments supplied in this field only if you require special installer behavior.

Wizard Element	Element Type	Comments
Major Version	Text field	Enter a value from the first part of the release number. For example, for Siebel CRM version 17.0, set Major Version to 17.
Minor Version	Text field	Enter a value from the second part of the release number. For example, for Siebel CRM version 17.0, set Minor Version to 0.
Maintenance Version	Text field	Enter a value from the third part of the release number. For example, for Siebel CRM version 17.0, set Maintenance Version to 0.
Patch Version	Text field	Enter a value from the fourth part of the release number. For example, for Siebel CRM version 17.0, set Patch Version to 0.
Vertical Code	Drop-down list	Select SIA for Siebel Industry Applications.
Activate Upgrade Kit	Check box	It is recommended that you clear this check box for all Siebel Client Executables upgrade kits. If this check box is selected, then the files to be included in the upgrade kit will be compressed into a single archive on the Siebel File System automatically. You can also perform activation manually, as described in *Activating an Upgrade Kit*. For most upgrade kits, it is recommended that you select this check box. However, if you are creating an upgrade kit that is dependent on another upgrade kit, then make sure to clear this check box. For more information about working with dependent upgrade kits, see *Controlling the Order of Upgrade Kit Installation*.
Upgrade Kit Title	Text field	An identifier for the upgrade kit. The default value is the value of Upgrade Component Name followed by a space and the value of New Version. However, you can modify the title while defining the upgrade kit. After the upgrade kit is defined, this identifier appears in the Name field of the Upgrade Kits list.
Apply Versions	Check box	This check box is displayed only if you previously selected the Activate Upgrade Kit check box in the Upgrade Kit Wizard. It is recommended that you clear this check box for all Siebel Client Executables upgrade kits. When this check box is selected, the compiled information string in the database will be updated automatically with the component version information for this upgrade kit. You can also apply versions manually, as described in *Applying an Upgrade Kit*. For most upgrade kits, it is recommended that you select this check box. However, if you are creating an upgrade kit that is dependent on another upgrade kit, then make sure to clear this check box. For more information about working with dependent upgrade kits, see *Controlling the Order of Upgrade Kit Installation*.
Required Upgrade Kit	Check box	This check box is displayed only if you previously selected the Apply Versions check box in the Upgrade Kit Wizard.

Wizard Element	Element Type	Comments
		When this check box is selected, the upgrade kit will be required regardless of previous versions installed. (Min Version and Max Version are set equal to the value of New Version, but this change is not visible in the Upgrade Kit Wizard.)
Description	Text field	This field is available for comments about the upgrade kit. Comments entered here are displayed in the Upgrade Kits list.

Note: To avoid unnecessary download operations, it is recommended that you keep only one upgrade kit for history-independent components, including Siebel Client Executables upgrade kits. Depending on your preference, you can either delete or deactivate previous upgrade kits for a history-independent component. Deleting an upgrade kit increases available space in the Siebel File System. Deactivating an upgrade kit prevents use of the upgrade kit, while keeping it available in case you need it unexpectedly.

Roadmap for Distributing a Siebel Innovation Pack or Patchset Release

This topic provides an overview of the process of deploying a Siebel Innovation Pack release or Siebel Patchset release, including links to more specific instructions.

Note: Only Siebel Mobile and Developer Web Clients can use Siebel Anywhere to download the upgrade kits needed for a Siebel Innovation Pack release or Siebel Patchset release. For other types of clients, such as Siebel Tools clients, users can only receive notification and then manually install the release. For more information about manually installing the latest Siebel Innovation Pack release, see *Siebel Installation Guide* . For information about manually installing the latest Siebel Patchset release, see *Siebel Patchset Installation Guide for Siebel CRM*.

To distribute a Siebel Innovation Pack release or Siebel Patchset release, perform the following tasks or processes:

1. Determine your upgrade requirements. For more information about this step, see *Determining Your Upgrade Requirements*.
2. Prepare any needed infrastructure elements, such as configurations or custom components. For more information about this step, see *Modifying and Creating Siebel Anywhere Configurations* and *Setting Up Custom Siebel Anywhere Upgrade Components*.
3. Prepare Siebel Innovation Pack release files or Siebel Patchset release files for inclusion in upgrade kits. Check the release-specific documentation for any special requirements concerning this step.

   Note: If you want to create a delta installation upgrade kit, rather than a standard installation upgrade kit, then this is the point at which you create the delta patch file. For more information about delta installation upgrade kits, see *About Reducing the Size of Siebel Client Executables Upgrade Kits* and *Roadmap for Creating a Delta Installation Siebel Client Executables Upgrade Kit*.

4. Define one or more upgrade kits, as needed:
    a. Define a single standard installation upgrade kit for the language-independent (base) portion of the Siebel Innovation Pack release or Siebel Patchset release and for any language-specific portions. For more information about this step, see *Defining a Siebel Client Executables Upgrade Kit*.
    b. Or, define delta installation upgrade kits. For more information about delta installation upgrade kits, see *About Reducing the Size of Siebel Client Executables Upgrade Kits* and *Roadmap for Creating a Delta Installation Siebel Client Executables Upgrade Kit*.
5. Activate the upgrade kit you created. For more information about this step, see *Activating an Upgrade Kit*.
6. Apply the upgrade kit you created. For more information about this step, see *Applying an Upgrade Kit*.
7. Distribute the upgrade kit to a test configuration. For more information about this step, see *Distributing Upgrade Kits*.
8. Test the upgrade kit by retrieving and installing it as both local and remote members of the test configuration, and by running the software that has been upgraded. For more information about this step, see *Retrieving, Installing, and Testing Upgrade Kits*
9. After the upgrade kit is operating correctly for test configuration members, distribute the upgrade kit to one or more additional configurations, for general use.

Related Books

Siebel Installation Guide

Siebel Patchset Installation Guide for Siebel CRM.

Roadmap for Creating a Delta Installation Siebel Client Executables Upgrade Kit

Siebel Anywhere supports the use of delta installation Siebel Client Executables upgrade kits. For more information about this type of upgrade kit, see *About Reducing the Size of Siebel Client Executables Upgrade Kits*.

To create a delta installation client executables upgrade kit, perform the following tasks:

1. Consult the release-specific documentation for the release you want to install, to determine whether you must define an Upgrade Wizard upgrade kit. If so, prepare and define the Upgrade Wizard upgrade kit, as described in the following topics:
    - *Preparing Contents for a Siebel Upgrade Wizard Upgrade Kit*
    - *Defining a Siebel Upgrade Wizard Upgrade Kit*
2. Identify the computer where you will create the delta patch file. This is typically a Siebel administrator's computer. Install the version of the Siebel Mobile Web Client software that your mobile users *currently* use, if that version of the software is not already present.
3. On the same computer, but in a different directory, install the client software for the new release.

    > **Note:** The client environment is sufficient for creating the delta patch file; no servers are required at this time.

4. Create a delta patch file for each of the following components, as described in *Creating a Delta Patch File*:

- Siebel Client Executables base (language-independent) component
- (Optional) Each Siebel Client Executables__*language-code* component that is in use in your Siebel Business Applications implementation

5. Define a Siebel Client Executables upgrade kit for each of the components listed in Step *Roadmap for Creating a Delta Installation Siebel Client Executables Upgrade Kit*, choosing Delta Install for the Select Type of Installation setting in the Upgrade Kit Wizard.

   For more information about defining a Siebel Client Executables upgrade kit, see *Defining a Siebel Client Executables Upgrade Kit*.
6. If you defined an Upgrade Wizard upgrade kit, then make the Siebel Client Executables base upgrade kit dependent on the Upgrade Wizard upgrade kit, and make each Siebel Client Executables__*language-code* upgrade kit dependent on the base upgrade kit.

   For more information about making one upgrade kit dependent upon another, see *Controlling the Order of Upgrade Kit Installation*.

Defining a Siebel Patchset Upgrade Kit

This topic provides specific instructions for defining a Siebel Patchset upgrade kit using the Upgrade Kit Wizard. Instructions in this topic apply to the Siebel Quick Fix component type, which is used for delivering Siebel Patchset releases to Siebel clients. Siebel Patchset releases are language-independent.

> **Note:** For Siebel Innovation Pack releases, use Siebel Client Executables upgrade kits, as described in *Defining a Siebel Client Executables Upgrade Kit*.

The following procedure describes how to define a Siebel Patchset upgrade kit.

To define a Siebel Patchset upgrade kit

1. Gather the information and files you will need, as described in *Planning and Preparing to Use Siebel Anywhere* .
2. Start the Upgrade Kit Wizard, as described in *Running the Upgrade Kit Wizard*.
3. Use the information you have gathered in the previous step to respond to the prompts in the Upgrade Kit Wizard.

   > **Note:** Minimum Old Version and Maximum Old Version are automatically set to NULL for the first upgrade kit for the Siebel Quick Fix component. Specify New Version manually. For subsequent upgrade kids, Maximum Old Version is automatically set to the previous value of New Version and Minimum Old Version is set to NULL. Again, specify New Version manually. These settings are displayed in the final screen of the Upgrade Kit Wizard.

4. When you have finished specifying data about the upgrade kit you are defining, click Finish to pass the request to the Upgrade Kit Builder server component and to exit from the Upgrade Kit Wizard.

   A new row for the new upgrade kit appears in the Upgrade Kits list, with Status set to Request Submitted. For more information about the upgrade kit information available in this list and other lists, see *Viewing Upgrade Kit Properties*.

   After you define your upgrade kit, you must proceed to activate, apply, and distribute it. For more information about these tasks, see *Activating, Applying, and Distributing Upgrade Kits*.

The following information is presented in approximately the sequence used by the Upgrade Kit Wizard.

Wizard Element	Element Type	Comments
Upgrade Component	Drop-down list	The name of the component the upgrade kit will install or upgrade. Select Siebel Quick Fix.
UNC Path for File Directory	Text field	The Universal Naming Convention (UNC) path to the directory that contains the files that you prepared for the upgrade, as described in *Preparing Contents for a Siebel Client Executables Upgrade Kit*. Identify a network-accessible location. An example path is: \\ computer_name **\siebel\patchset**
Destination Directory	Text field	Specifies where the files of the upgrade kit are to be placed on the client computer. It is strongly recommended to use the default value.
Delete destination file when done	Check box	When this check box is selected, the upgrade kit files will be deleted, automatically, after the upgrade kit is installed.
Specify file to execute	Text field	Specifies one file in the upgrade kit to be executed as part of installation of the upgrade kit. This setting is automatically set to siebopatch.bat, indicating that siebopatch.bat will be run from the directory identified in UNC Path for File Directory. (This file is automatically generated by the Upgrade Kit Wizard.)
Enter command line arguments	Text field	Specifies any command-line arguments to be used by the Siebel Patchset installer executable program. The default value is: **$SiebelRoot $SiebelRoot\\Upgrade\\$KitName apply -verbose**

Wizard Element	Element Type	Comments
		where $SiebelRoot refers to the Siebel client root directory, and $KitName refers to a subdirectory with the name of the upgrade kit. Both of these variables are case-sensitive. In general, change the default command-line arguments supplied in this field only if you require special installer behavior.
Acceptable Return Code	Text field	An optional setting that specifies the numeric code that the specified file to execute returns upon successful execution. The default value is 0 (zero). You can also specify multiple comma-separated values. After the file is executed, the Upgrade Wizard checks the actual return code against the value you specify in this setting.
Upgrade Kit Title	Text field	An identifier for the upgrade kit. The default value is the value of Upgrade Component Name followed by a space and the value of New Version. However, you can modify the title while defining the upgrade kit. After the upgrade kit is defined, this identifier appears in the Name field of the Upgrade Kits list.
Minimum Old Version	Read-only text field	The earliest component version that can download and install the upgrade kit. This setting is automatically set to NULL, indicating that there are no prerequisite versions required for using the upgrade kit. For more information about choosing version values, see *Determining Version-Setting Values*.
Maximum Old Version	Read-only text field	The latest component version that can download and install the upgrade kit. This setting is automatically set to NULL for the first upgrade kit. For subsequent upgrade kits, it is set to the previous value of New Version, so that an upgrade kit will not be installed over a newer version. For more information about choosing version values, see *Determining Version-Setting Values*.
New Version	Text field	The version number that the component being upgraded will have after the upgrade kit is installed. The default value is 1 greater than the current version for the component to be upgraded. The New Version value *must* be an integer. For more information about choosing version values, see *Determining Version-Setting Values*.
Activate Upgrade Kit	Check box	Clear this check box for all Siebel Patchset upgrade kits.

Wizard Element	Element Type	Comments
		If this check box is selected, then the files to be included in the upgrade kit will be compressed into a single archive on the Siebel File System automatically. You can also perform activation manually, as described in *Activating, Applying, and Distributing Upgrade Kits* For more information about working with dependent upgrade kits, see *Controlling the Order of Upgrade Kit Installation*.
Description	Text field	This field is available for comments about the upgrade kit. Comments entered here are displayed in the Upgrade Kits list.

Note: To avoid unnecessary download operations, it is recommended that you delete or deactivate previous upgrade kits for the Siebel Quick Fix component. Deleting an upgrade kit increases available space in the Siebel File System. Deactivating an upgrade kit prevents use of the upgrade kit, while keeping it available in case you need it unexpectedly.

Defining a Siebel Repository Export Import Upgrade Kit

This topic provides specific instructions for defining a Siebel Repository Export Import upgrade kit using the Upgrade Kit Wizard.

To define a Siebel Repository Import Export upgrade kit

1. Gather the information and files you will need, as described in *Planning and Preparing to Use Siebel Anywhere*
2. Start the Upgrade Kit Wizard, as described in *Running the Upgrade Kit Wizard*.
3. Use the information you have gathered in the previous step to respond to the prompts in the Upgrade Kit Wizard.
4. When you have finished specifying data about the upgrade kit you are defining, click Finish to pass the request to the Upgrade Kit Builder server component and to exit from the Upgrade Kit Wizard.

   A new row for the new upgrade kit appears in the Upgrade Kits list, with Status set to Request Submitted. For more information about the upgrade kit information available in this list and other lists, see *Viewing Upgrade Kit Properties*.

   After you define your upgrade kit, you must proceed to activate, apply, and distribute it. For more information about these tasks, see *Activating, Applying, and Distributing Upgrade Kits* .

The following information is presented in approximately the sequence used by the Upgrade Kit Wizard.

Wizard Element	Element Type	Comments
ODBC Data Source	Text field	The name of the ODBC data source used to connect to the HQ database. This is a required field.
User Name	Text field	The database user name. This is a required field.
User Password	Text field	The database user password. This is a required field.
Schema Qualifier	Text field	The database Table owner. This is a required field.
Repository Name	Read-only text field	The value in this field defaults to Siebel Repository.
Minimum Old Version	Read-only text field	The earliest component version that can download and install the upgrade kit. This setting is automatically set to NULL for history-independent component types, including repository files, indicating that there are no prerequisite versions required for using the upgrade kit. For more information about choosing version values, see *Determining Version-Setting Values*.
Maximum Old Version	Read-only text field	The latest component version that can download and install the upgrade kit. This setting is automatically set to NULL for history-independent component types, including repository files, indicating that there are no prerequisite versions required for using the upgrade kit. For more information about choosing version values, see *Determining Version-Setting Values*.
New Version	Text field	The version number that the component being upgraded will have after the upgrade kit is installed. The default value is 1 greater than the current version for the component to be upgraded. For a Siebel Repository Export Import upgrade kit, the New Version value *must* be an integer. For more information about choosing version values, see *Determining Version-Setting Values*.
Activate Upgrade Kit	Check box	When this check box is selected, the files to be included in the upgrade kit will be compressed into a single archive on the Siebel File System automatically. You can also perform activation manually, as described in *Activating, Applying, and Distributing Upgrade Kits*

Wizard Element	Element Type	Comments
		For most upgrade kits, it is recommended that you select this check box. However, if you are creating an upgrade kit that is dependent on another upgrade kit, then clear this check box. (For example, a repository upgrade kit might be dependent on a Siebel Database Schema upgrade kit.) For more information about working with dependent upgrade kits, see *Controlling the Order of Upgrade Kit Installation*.
Apply Versions	Check box	This check box is displayed only if you previously selected the Activate Upgrade Kit check box in the Upgrade Kit Wizard. When this check box is selected, the compiled information string in the database will be updated automatically with the component version information for this upgrade kit. You can also apply versions manually, as described in *Activating, Applying, and Distributing Upgrade Kits* For most upgrade kits, it is recommended that you select this check box. However, if you are creating an upgrade kit that is dependent on another upgrade kit, then clear this check box. For more information about working with dependent upgrade kits, see *Controlling the Order of Upgrade Kit Installation*.
Required Upgrade Kit	Check box	This check box is displayed only if you previously selected the Apply Versions check box in the Upgrade Kit Wizard. When this check box is selected, the upgrade kit will be required regardless of previous versions installed. (Min Version and Max Version are set equal to the value of New Version, but this change is not visible in the Upgrade Kit Wizard.)
Upgrade Kit Title	Text field	An identifier for the upgrade kit. The default value is the value of Upgrade Component Name followed by a space and the value of New Version. However, you can modify the title while defining the upgrade kit. After the upgrade kit is defined, this identifier appears in the Name field of the Upgrade Kits list.
Comments	Text field	This field is available for comments about the upgrade kit. Comments entered here are displayed in the Upgrade Kits list.

Defining a Third Party Software Upgrade Kit

This topic provides specific instructions for defining a Third Party Software upgrade kit using the Upgrade Kit Wizard.

For an example of the creation of a Third Party Software upgrade kit, see *Example of Constructing a Third Party Software Upgrade Kit*.

> **Note:** Third Party Software components are meant to support third-party software products that are needed for operation with Siebel applications. Siebel Anywhere is not a general-purpose tool for software configuration management and distribution.

To define a Third Party Software upgrade kit

1. Gather the information and files you will need, as described in *Planning and Preparing to Use Siebel Anywhere*.
2. Start the Upgrade Kit Wizard, as described in *Running the Upgrade Kit Wizard*.
3. Use the information you have gathered in the previous step to respond to the prompts in the Upgrade Kit Wizard.
4. When you have finished specifying data about the upgrade kit you are defining, click Finish to pass the request to the Upgrade Kit Builder server component and to exit from the Upgrade Kit Wizard.

   A new row for the new upgrade kit appears in the Upgrade Kits list, with Status set to Request Submitted. For more information about the upgrade kit information available in this list and other lists, see *Viewing Upgrade Kit Properties*.

   After you define your upgrade kit, you must proceed to activate, apply, and distribute it. For more information about these tasks, see *Activating, Applying, and Distributing Upgrade Kits*.

The following information is presented in approximately the sequence used by the Upgrade Kit Wizard.

Wizard Element	Element Type	Comments
Upgrade Component	Drop-down list	The name of the component the upgrade kit will install or upgrade. Select one of the following predefined components, or a custom component you have created: Third Party - Oracle 8 Client Third Party - Microsoft Word Third Party - Microsoft SQL Server Driver Third Party - Microsoft Internet Explorer Third Party - IBM DB2 Client Third Party - Adobe Acrobat Reader

Wizard Element	Element Type	Comments
		Third Party - Adobe Acrobat
UNC Path for Software Directory	Text field	The Universal Naming Convention (UNC) path to the directory that contains the files to include in the upgrade kit. All files in the specified directory are included in the upgrade kit automatically.
Destination Directory	Text field	Specifies where the files from the upgrade kit are to be placed on the client computer. The default value is: `$SiebelRoot\upgrade\$KitName` where $SiebelRoot refers to the Siebel client root directory, and $KitName refers to a subdirectory with the name of the upgrade kit. Both of these variables are case-sensitive.
Delete destination file when done	Check box	When this check box is selected, the upgrade kit files will be deleted, automatically, after the upgrade kit is installed.
Specify file to execute	Text field	Specifies the file that runs the software installer on the subscriber's computer, such as install.exe. Only one file for each upgrade kit can be specified for execution. Make sure that you specify a file that can be executed, such as a BAT or EXE file on Windows. If you specify a file that *cannot* be executed, such as a TXT file or a CFG file, then the upgrade kit will not install properly.
Enter command line arguments	Text field	Specifies any command-line arguments to use when running the file chosen in the Specify file to execute field.
Minimum Old Version	Text field	The earliest component version that can download and install the upgrade kit. When this field is blank, this indicates that there are no prerequisite versions required for using the upgrade kit. The default value is the current component version recorded in the database. For more information about choosing version values, see *Determining Version-Setting Values*.
Maximum Old Version	Text field	The latest component version that can download and install the upgrade kit. When this field is blank, this indicates that there are no prerequisite versions required for using the upgrade kit. For more information about choosing version values, see *Determining Version-Setting Values*.

Wizard Element	Element Type	Comments
New Version	Text field	The version number that the component being upgraded will have after the upgrade kit is installed. The default value is 1 greater than the current version for the component to be upgraded. For more information about choosing version values, see *Determining Version-Setting Values*.
Activate Upgrade Kit	Check box	When this check box is selected, the files to be included in the upgrade kit will be compressed into a single archive on the Siebel File System automatically. You can also perform activation manually, as described in *Activating, Applying, and Distributing Upgrade Kits* For most upgrade kits, it is recommended that you select this check box. However, if you are creating an upgrade kit that is dependent on another upgrade kit, then clear this check box. For more information about working with dependent upgrade kits, see *Controlling the Order of Upgrade Kit Installation*.
Apply Versions	Check box	This check box is displayed only if you previously selected the Activate Upgrade Kit check box in the Upgrade Kit Wizard. When this check box is selected, the compiled information string in the database will be updated automatically with the component version information for this upgrade kit. You can also apply versions manually, as described in *Activating, Applying, and Distributing Upgrade Kits* For most upgrade kits, it is recommended that you select this check box. However, if you are creating an upgrade kit that is dependent on another upgrade kit, then clear this check box. For more information about working with dependent upgrade kits, see *Controlling the Order of Upgrade Kit Installation*.
Required Upgrade Kit	Check box	This check box is displayed only if you previously selected the Apply Versions check box in the Upgrade Kit Wizard. When this check box is selected, the upgrade kit will be required regardless of previous versions installed. (Min Version and Max Version are set equal to the value of New Version, but this change is not visible in the Upgrade Kit Wizard.)

Wizard Element	Element Type	Comments
Upgrade Kit Title	Text field	An identifier for the upgrade kit. The default value is the value of Upgrade Component Name followed by a space and the value of New Version. However, you can modify the title while defining the upgrade kit. After the upgrade kit is defined, this identifier appears in the Name field of the Upgrade Kits list.
Comments	Text field	This field is available for comments about the upgrade kit. Comments entered here are displayed in the Upgrade Kits list.

Defining a Siebel Workspace Seed Import

This topic provides specific instructions for defining a Siebel Workspace Seed Import using the Upgrade Kit Wizard.

To define a Siebel Workspace Seed Import

1. Gather the information and files you will need, as described in *Planning and Preparing to Use Siebel Anywhere*.
2. Start the Upgrade Kit Wizard, as described in *Running the Upgrade Kit Wizard*.
3. Use the information you have gathered in the previous step to respond to the prompts in the Upgrade Kit Wizard.
4. When you have finished specifying data about the upgrade kit you are defining, click Finish to pass the request to the Upgrade Kit Builder server component and to exit from the Upgrade Kit Wizard.

   A new row for the new upgrade kit appears in the Upgrade Kits list, with Status set to Request Submitted. For more information about the upgrade kit information available in this list and other lists, see *Viewing Upgrade Kit Properties*.

   After you define your upgrade kit, you must proceed to activate, apply, and distribute it. For more information about these tasks, see *Activating, Applying, and Distributing Upgrade Kits*.

The following information is presented in approximately the sequence used by the Upgrade Kit Wizard.

Wizard Element	Element Type	Comments
ODBC Data Source	Text field	The name of the ODBC data source used to connect to the HQ database. This is a required field.
User Name	Text field	The database user name. This is a required field.
User Password	Text field	The database user password. This is a required field.

Wizard Element	Element Type	Comments
Schema Qualifier	Text field	The database Table owner. This is a required field.
Repository Name	Read-only text field	The value in this field defaults to Siebel Repository.
Upgrade Component	Drop-down list	The name of the component the upgrade kit will install or upgrade. Select Siebel Workspace Seed Import.
Select File Uploading Method	Radio button	The method for specifying files to include in the upgrade kit. Select one of the following: **Add Files.** Use when the number of files to be added is small. **Use UNC Path.** Use when the number of files to be added is large. Gather all the files to be added into one directory before proceeding, as you will specify the directory in a later screen of the wizard.
Files to Add	Text field	This field is displayed only if you previously selected the Add Files radio button in the Upgrade Kit Wizard. The names of files to include in the upgrade kit. This field is populated either by entering one file name and path, or by clicking Browse and choosing a file from the directory listings displayed. In either case, click the Add button after specifying each path and file name combination. Repeat to add more files.
Browse	Button	This button is displayed only if you previously selected the Add Files radio button in the Upgrade Kit Wizard. This button displays a standard dialog box for browsing and choosing a file.
Add	Button	This button is displayed only if you previously selected the Add Files radio button in the Upgrade Kit Wizard. When this button is clicked, the information in the Files To Add field is saved. If the Add button is not clicked, then the information is discarded when you click Next, and an error message is displayed.

Wizard Element	Element Type	Comments
Added Files	Drop-down list	This list is displayed only if you previously selected the Add Files radio button in the Upgrade Kit Wizard. This list displays the names of the files added to the upgrade kit so far. This setting is automatically populated when you enter information in Files to Add and then click the Add button.
Remove	Button	This button is displayed only if you previously selected the Add Files radio button in the Upgrade Kit Wizard. When this button is clicked, the Upgrade Kit Wizard discards the file information currently selected and displayed in the Added Files drop-down list. Other items in the drop-down list are not affected.
UNC Path for File Directory	Text field	This field is displayed only if you previously selected the Use UNC Path radio button in the Upgrade Kit Wizard. This field specifies the Universal Naming Convention (UNC) path of a directory that contains all the files to include in the upgrade kit. All the files in the directory you specify will be included in the upgrade kit.
Destination Directory	Text field	This field specifies where the files of the upgrade kit are to be placed on the client computer.
Delete destination file when done	Check box	When this check box is selected, the upgrade kit files will be deleted, automatically, after the upgrade kit is installed.
Select file to execute	Drop-down list	Specifies one file in the upgrade kit to be executed as part of the installation of the upgrade kit. Available values are the values stored in the Added Files field. Make sure that you specify a file that can be executed, such as a BAT or EXE file. If you specify a file that *cannot* be executed, such as a TXT file or a CFG file, then the upgrade kit will not install properly.
Enter command line arguments	Text field	Specifies any command-line arguments to use when running the file chosen in the Select file to execute drop-down list.

Wizard Element	Element Type	Comments
Acceptable Return Code	Numeric field	An optional setting that specifies the numeric code that the specified file to execute returns upon successful execution. The default value is 0 (zero). You can also specify multiple comma-separated values. After the file is executed, the Upgrade Wizard checks the actual return code against the value you specify in this setting.
Minimum Old Version	Text field	The earliest component version that can download and install the upgrade kit. When this field is blank, this indicates that there are no prerequisite versions required for using the upgrade kit. The default value is the current component version recorded in the database. For more information about choosing version values, see *Determining Version-Setting Values*.
Maximum Old Version	Text field	The latest component version that can download and install the upgrade kit. When this field is blank, this indicates that there are no prerequisite versions required for using the upgrade kit. For more information about choosing version values, see *Determining Version-Setting Values*.
New Version	Text field	The version number that the component being upgraded will have after the upgrade kit is installed. The default value is 1 greater than the current version for the component to be upgraded. For more information about choosing version values, see *Determining Version-Setting Values*.
Activate Upgrade Kit	Check box	When this check box is selected, the files to be included in the upgrade kit will be compressed into a single archive on the Siebel File System automatically. You can also perform activation manually, as described in *Activating, Applying, and Distributing Upgrade Kits* . For most upgrade kits, it is recommended that you select this check box. However, if you are creating an upgrade kit that is dependent on another upgrade kit, then make sure to clear this check box. For more information about working with dependent upgrade kits, see *Controlling the Order of Upgrade Kit Installation*.
Apply Versions	Check box	This check box is displayed only if you previously selected the Activate Upgrade Kit check box in the Upgrade Kit Wizard.

Wizard Element	Element Type	Comments
		When this check box is selected, the compiled information string in the database will be updated automatically with the component version information for this upgrade kit. You can also apply versions manually, as described in *Activating, Applying, and Distributing Upgrade Kits* For most upgrade kits, it is recommended that you select this check box. However, if you are creating an upgrade kit that is dependent on another upgrade kit, then clear this check box. For more information about working with dependent upgrade kits, see *Controlling the Order of Upgrade Kit Installation*.
Required Upgrade Kit	Check box	This check box is displayed only if you previously selected the Apply Versions check box in the Upgrade Kit Wizard. When this check box is selected, the upgrade kit will be required regardless of previous versions installed. (Min Version and Max Version are set equal to the value of New Version, but this change is not visible in the Upgrade Kit Wizard.)
Upgrade Kit Title	Text field	An identifier for the upgrade kit. The default value is the value of Upgrade Component Name followed by a space and the value of New Version. However, you can modify the title while defining the upgrade kit. After the upgrade kit is defined, this identifier appears in the Name field of the Upgrade Kits list.
Comments	Text field	This field is available for comments about the upgrade kit. Comments entered here are displayed in the Upgrade Kits list.

Defining a Siebel Client Customer Revisions Upgrade Kit

This topic provides specific instructions for defining a Siebel Client Customer Revisions (of type Customer Revision) upgrade kit using the Upgrade Kit Wizard. These instructions can be used for the following types of upgrade components, where (base) indicates a language-independent component, and *language-code* indicates a component that applies to a particular language represented by a code such as ENU (U.S. English) or DEU (German):

- Siebel Client Customer Revisions (base)

- Siebel Client Customer Revisions__*language-code*

These instructions can also be used for custom components.

CAUTION: Do not use the upgrade kit type Customer Revision for upgrade kits containing a Siebel runtime repository data file or CFG file. (Using a Customer Revision upgrade kit type to deliver a Siebel runtime repository data file or a CFG file can result in upgrade failures or incorrect version checking and version numbering.) Instead, use the Siebel Repository Export Import and Siebel Client CFG upgrade kit types, respectively, for these upgrades.

To define a Siebel Client Customer Revisions upgrade kit

1. Gather the information and files you will need, as described in *Planning and Preparing to Use Siebel Anywhere*.
2. Start the Upgrade Kit Wizard, as described in *Running the Upgrade Kit Wizard*.
3. Use the information you have gathered previously step to respond to the prompts in the Upgrade Kit Wizard.
4. When you have finished specifying data about the upgrade kit you are defining, click Finish to pass the request to the Upgrade Kit Builder server component and to exit from the Upgrade Kit Wizard.

   A new row for the new upgrade kit appears in the Upgrade Kits list, with Status set to Request Submitted. For more information about the upgrade kit information available in this list and other lists, see *Viewing Upgrade Kit Properties*.

   After you define your upgrade kit, you must proceed to activate, apply, and distribute it. For more information about these tasks, see *Activating, Applying, and Distributing Upgrade Kits*.

The following information is presented in approximately the sequence used by the Upgrade Kit Wizard.

Wizard Element	Element Type	Comments
Upgrade Component	Drop-down list	The name of the component the upgrade kit will install or upgrade. Select one of the following: Siebel Client Customer Revisions Siebel Client Customer Revisions__*language-code* For a custom component, select the name of the component you created.
Select File Uploading Method	Radio button	The method for specifying files to include in the upgrade kit. Select one of the following: **Add Files.** Use when the number of files to be added is small. **Use UNC Path.** Use when the number of files to be added is large. Gather all the files to be added into one directory before proceeding, as you will specify the directory in a later screen of the wizard.

Wizard Element	Element Type	Comments
Files to Add	Text field	This field is displayed only if you previously selected the Add Files radio button in the Upgrade Kit Wizard. The names of files to include in the upgrade kit. This field is populated either by entering one file name and path, or by clicking Browse and choosing a file from the directory listings displayed. In either case, click the Add button after specifying each path and file name combination. Repeat to add more files.
Browse	Button	This button is displayed only if you previously selected the Add Files radio button in the Upgrade Kit Wizard. This button displays a standard dialog box for browsing and choosing a file.
Add	Button	This button is displayed only if you previously selected the Add Files radio button in the Upgrade Kit Wizard. When this button is clicked, the information in the Files To Add field is saved. If the Add button is not clicked, then the information is discarded when you click Next, and an error message is displayed.
Added Files	Drop-down list	This list is displayed only if you previously selected the Add Files radio button in the Upgrade Kit Wizard. This list displays the names of the files added to the upgrade kit so far. This setting is automatically populated when you enter information in Files to Add and then click the Add button.
Remove	Button	This button is displayed only if you previously selected the Add Files radio button in the Upgrade Kit Wizard. When this button is clicked, the Upgrade Kit Wizard discards the file information currently selected and displayed in the Added Files drop-down list. Other items in the drop-down list are not affected.
UNC Path for File Directory	Text field	This field is displayed only if you previously selected the Use UNC Path radio button in the Upgrade Kit Wizard. This field specifies the Universal Naming Convention (UNC) path of a directory that contains all the files to

Wizard Element	Element Type	Comments
		include in the upgrade kit. All the files in the directory you specify will be included in the upgrade kit.
Destination Directory	Text field	This field specifies where the files of the upgrade kit are to be placed on the client computer.
Delete destination file when done	Check box	When this check box is selected, the upgrade kit files will be deleted, automatically, after the upgrade kit is installed.
Select file to execute	Drop-down list	Specifies one file in the upgrade kit to be executed as part of the installation of the upgrade kit. Available values are the values stored in the Added Files field. Make sure that you specify a file that can be executed, such as a BAT or EXE file. If you specify a file that *cannot* be executed, such as a TXT file or a CFG file, then the upgrade kit will not install properly.
Enter command line arguments	Text field	Specifies any command-line arguments to use when running the file chosen in the Select file to execute drop-down list.
Acceptable Return Code	Numeric field	An optional setting that specifies the numeric code that the specified file to execute returns upon successful execution. The default value is 0 (zero). You can also specify multiple comma-separated values. After the file is executed, the Upgrade Wizard checks the actual return code against the value you specify in this setting.
Minimum Old Version	Text field	The earliest component version that can download and install the upgrade kit. When this field is blank, this indicates that there are no prerequisite versions required for using the upgrade kit. The default value is the current component version recorded in the database. For more information about choosing version values, see *Determining Version-Setting Values*.
Maximum Old Version	Text field	The latest component version that can download and install the upgrade kit. When this field is blank, this indicates that there are no prerequisite versions required for using the upgrade kit. For more information about choosing version values, see *Determining Version-Setting Values*.

Wizard Element	Element Type	Comments
New Version	Text field	The version number that the component being upgraded will have after the upgrade kit is installed. The default value is 1 greater than the current version for the component to be upgraded. For more information about choosing version values, see *Determining Version-Setting Values*.
Activate Upgrade Kit	Check box	When this check box is selected, the files to be included in the upgrade kit will be compressed into a single archive on the Siebel File System automatically. You can also perform activation manually, as described in *Activating, Applying, and Distributing Upgrade Kits* For most upgrade kits, it is recommended that you select this check box. However, if you are creating an upgrade kit that is dependent on another upgrade kit, then make sure to clear this check box. For more information about working with dependent upgrade kits, see *Controlling the Order of Upgrade Kit Installation*.
Apply Versions	Check box	This check box is displayed only if you previously selected the Activate Upgrade Kit check box in the Upgrade Kit Wizard. When this check box is selected, the compiled information string in the database will be updated automatically with the component version information for this upgrade kit. You can also apply versions manually, as described in *Activating, Applying, and Distributing Upgrade Kits* . For most upgrade kits, it is recommended that you select this check box. However, if you are creating an upgrade kit that is dependent on another upgrade kit, then clear this check box. For more information about working with dependent upgrade kits, see *Controlling the Order of Upgrade Kit Installation*.
Required Upgrade Kit	Check box	This check box is displayed only if you previously selected the Apply Versions check box in the Upgrade Kit Wizard. When this check box is selected, the upgrade kit will be required regardless of previous versions installed. (Min Version and Max Version are set equal to the value of New Version, but this change is not visible in the Upgrade Kit Wizard.)

Wizard Element	Element Type	Comments
Upgrade Kit Title	Text field	An identifier for the upgrade kit. The default value is the value of Upgrade Component Name followed by a space and the value of New Version. However, you can modify the title while defining the upgrade kit. After the upgrade kit is defined, this identifier appears in the Name field of the Upgrade Kits list.
Comments	Text field	This field is available for comments about the upgrade kit. Comments entered here are displayed in the Upgrade Kits list.

Defining a Custom Component Upgrade Kit

The instructions for defining a custom component upgrade kit depend on the component type you choose when you create the custom component. For more information about which instructions to use for each component type, see the following table.

If Custom Component Type Is	Then Use the Following Instructions
Siebel CFG File	*Defining a Siebel Configuration File (CFG) Upgrade Kit*
Siebel Database Extension	*Defining a Siebel Database Schema Upgrade Kit*
Siebel Client Executables	*Defining a Siebel Client Executables Upgrade Kit*
Siebel Quick Fix	*Defining a Siebel Patchset Upgrade Kit*
Siebel Repository Export Import	*Defining a Siebel Repository Export Import Upgrade Kit*
Third Party Software	*Defining a Third Party Software Upgrade Kit*
Customer Revision	*Defining a Siebel Client Customer Revisions Upgrade Kit*

Viewing Upgrade Kit Properties

After you use the Upgrade Kit Wizard to define an upgrade kit, information about the upgrade kit becomes visible in your Siebel application. Two general types of information are available:

- Information that applies to the upgrade kit as a whole, and the components that it will upgrade. See *Properties of the Entire Upgrade Kit and Its Components*.
- Information that applies to upgrade kit items and item parameters. See *Properties of Upgrade Kit Items and Upgrade Kit Item Parameters*.

Properties of the Entire Upgrade Kit and Its Components

The following procedure describes how to view and interpret information that applies to the upgrade kit as a whole and the components that it will upgrade.

To view the properties of an existing upgrade kit and its components

See the tables in this section for more information about:

- The fields in the Upgrade Kits list, which displays each upgrade kit defined in the Siebel Business Applications.
- How to interpret various values that are displayed in the Status field of this list.
- The fields in the Upgrade Kit Components list, which displays the components associated with the selected upgrade kit.

Field	Comments
Name	The name of the upgrade kit (called Upgrade Kit Title in the Upgrade Kit Wizard). The default name includes the component name and version in the upgrade kit. You can specify a different name when defining the upgrade kit. However, it is recommended that you use descriptive names if you do not accept the default wording. Also see *Identifying Configurations to Deliver Upgrade Components* for additional limitations for a name. The name cannot be modified after the upgrade kit has been defined.
Status	For descriptions of Status field values, see *Properties of the Entire Upgrade Kit and Its Components*.
Archive Kit Name	The name of the compressed file containing the upgrade kit.
File Size	The size of the physical file containing the upgrade kit.
Comments	This field is available for comments.
Modified	The date and time this record was last modified.

Field	Comments
Compiled Information	The compiled string that includes all information about the upgrade kit.

The Status field indicates whether the upgrade kit has the status of Request Submitted (that is, waiting for processing by the server), In Progress (in process of being created), Pending, Active, Error, and so on. *Properties of the Entire Upgrade Kit and Its Components* describes the values of the Status field.

Value	Description
Active	The upgrade kit is ready to be applied and distributed.
Error	An error occurred while the upgrade kit was being created.
Inactive	The upgrade kit is on hold and is not ready for further processing. The first step to activate the upgrade kit is to change its status to Pending. Then you can activate it.
In Progress	The creation of a new upgrade kit is underway on the server.
Pending	The upgrade kit was successfully created and is ready to be activated.
Request Submitted	The request has been submitted to the server to create the new upgrade kit.

The following table describes the fields in the Upgrade Kit Components list, which displays the components associated with the upgrade kit. The components are automatically created by the Upgrade Kit Wizard when you click Auto Create on the Upgrade Kits list.

When you select the new upgrade kit in the Upgrade Kit list, you will notice that two records appear in the Upgrade Kit Components list. The first record is the component to be upgraded. The second is the Siebel Upgrade Wizard. Including the Siebel Upgrade Wizard component reinforces using the correct version of the Upgrade Wizard for the upgrade.

CAUTION: Do not delete the default records in the Upgrade Kit Components list. These records store key information about how components work, and provide models that you might eventually need for setting up custom components.

Field Name	Comments
Name	The name of the component.
Min Old Version	The minimum component version that a subscriber must have to install this upgrade kit.

Field Name	Comments
Max Old Version	The maximum component version that a subscriber might have to install this upgrade kit.
New Version	The version of this component the user will have after installing the upgrade kit.
Comments	This field is available for comments.

The settings in the Upgrade Kit Components list also control the upgrade kit sequence (the order of upgrades), when multiple upgrade kits are used together. For example, if Upgrade Kit 1 must be installed before Upgrade Kit 2, then you can add component Upgrade Kit 1 upgrades as components of Upgrade Kit 2. For details, see *Controlling the Order of Upgrade Kit Installation*.

Properties of Upgrade Kit Items and Upgrade Kit Item Parameters

Each upgrade kit item defines an action in an upgrade kit and requires certain parameters to operate. For example, a file copy action needs the file to copy and the destination for the copy. The following procedure describes how to view and interpret information that applies to items and item parameters within a selected upgrade kit.

Note: It is recommended that you do not add, update, or delete any records in the Upgrade Kit Items list and Upgrade Kit Item Parameters list in the Upgrade Kit Items view. The records in these lists are created automatically, and changing them in any way can damage the selected upgrade kit.

To view properties of an existing upgrade kit's items and item parameters

- Navigate to the Administration - Siebel Anywhere screen, then the Upgrade Kit Items view.
  The Upgrade Kits list, the Upgrade Kit Items list, and the Upgrade Kit Item Parameters list appear.

For more information about the fields in the Upgrade Kits list, see the following table.

For more information about the fields in the Upgrade Kit Items list, which displays the individual items that define actions to be executed by a selected upgrade kit, see the following table.

For more information about the fields in the Upgrade Kit Item Parameters list, which displays the parameters associated with a selected upgrade kit item, see the following table.

Field Name	Comments
Comments	This field is available for comments.
Estimated Disk Space	The disk space required to download and execute an item (optional). When the upgrade kit is created, the default value for this field is 0. You can enter a new value (in bytes) before activating the upgrade kit.

Field Name	Comments
	Before a subscriber installs the upgrade kit, the Upgrade Wizard validates that the specified amount of free disk space is available. If adequate disk space is not available, then the Upgrade Wizard returns an error message.
Item Type	The type of item to execute.
Sequence	The order in which the Upgrade Wizard executes the item.
Title	The title of the item, which is displayed in the Upgrade Wizard.

Each upgrade kit item defines an action in an upgrade kit and requires certain parameters to operate. For example, a file copy action needs the file to copy and the destination for the copy. The Upgrade Kit Item Parameters list includes the parameters associated with each upgrade kit item. The following table describes the fields of this list.

Field Name	Comments
Attachment Name	The name of the file associated with this item.
Comments	This field is available for comments.
Date	The date and time when the file is attached.
File Ext	The extension of the file associated with this item.
Name	The name of the parameter.
Size	The size of the file associated with this item.
Value	The value of the parameter.

Example of Upgrade Kit Items and Upgrade Kit Item Parameters for Siebel Repository Export Import Component

The following table shows some example fields in the upgrade kit items list for the Siebel Repository Export Import component.

Sequence	Title	Item Type
1	Repository Data File	Repository Export Import
2	Importing Repository Changes	FileExecute
3	Copy File	FileCopy

The following table shows some example kit item parameters for Repository Data File:

Name	Attachment Name	File Ext	Description
Data File Name	Repexport	dat	new data

The following table shows some example kit item parameters for Importing Repository Changes:

Name	Value
File Name	$SiebelRoot\bin\repimexp
Command Line	/a p /u $UserName /p $Password /c "$ODBCDataSource" /D $TableOwner /R "$RepositoryName" /F "repexport.dat" /l $SiebelRoot\log\repimport.log /Y Mode=MWC
Check Return Code	Y
Return Code	0

The following table shows some example kit item parameters for FileCopy:

Name	Attachment Name	File Ext	Description
File Name	Repimp	cfg	new data
Destination	$SiebelRoot\bin\	None	None

5 Retrieving, Installing, and Testing Upgrade Kits

Retrieving, Installing, and Testing Upgrade Kits

This chapter discusses retrieving, installing, and testing upgrade kits. It includes the following topics:

- *Process of Retrieving, Installing, and Testing Upgrade Kits*
- *Retrieving and Installing Upgrade Kits*
- *Starting the Upgrade Wizard*
- *Retrieving Upgrade Kits for Siebel Regional Node Servers*
- *Testing Upgrade Kits*
- *Uninstalling a Delta Patch*

Process of Retrieving, Installing, and Testing Upgrade Kits

After you distribute an upgrade kit to a configuration, the subscribers associated with that configuration can retrieve and install that upgrade kit. It is strongly recommended that upgrade kits be retrieved, installed, and tested by test subscribers before you distribute them to production subscribers.

To retrieve, install, and test an upgrade kit, perform the following tasks:

1. *Retrieving and Installing Upgrade Kits*. The procedure for retrieving an upgrade kit depends on its type. Distribute upgrade kits to test subscribers before you distribute them to production subscribers.
2. *Starting the Upgrade Wizard*. Install the upgrade kit using the Upgrade Wizard. The process for invoking the Upgrade Wizard depends on the subscriber type. Have test subscribers install upgrade kits before you distribute them to production subscribers.
3. *Retrieving Upgrade Kits for Siebel Regional Node Servers*.
4. *Testing Upgrade Kits*. Have upgrade kits tested by test subscribers before you distribute them to production subscribers. (Among other possible problems, incorrectly constructed upgrade kits can prevent production users from writing data to Siebel Business Applications.)

If you encounter errors in the retrieval, installation, or testing process, then see the following for more information:

- *Starting the Upgrade Wizard* (information about client upgrade error recovery)
- *Retrieving Upgrade Kits for Siebel Regional Node Servers*

Retrieving and Installing Upgrade Kits

This topic describes how to retrieve and install upgrade kits. This task is a step in *Process of Retrieving, Installing, and Testing Upgrade Kits*.

The procedure for retrieving an upgrade kit depends on its type:

- Optional upgrade kits can be manually retrieved and installed by Siebel client subscribers through the Component Upgrades view.
- Required upgrade kits are automatically retrieved when mobile users synchronize or log into the application and when the server starts up or Replication Agent synchronizes with Siebel Server.

For each applicable Siebel client type, upgrade kits are retrieved as follows:

- Mobile Web Clients request the files from the Siebel Remote Server. The Remote Server retrieves the files from the Siebel File System.
- Developer Web Clients retrieve upgrade kits (.saf files) directly from the Siebel File System or by using the File System Manager, depending on the settings defined in the active CFG file.

Subscribers use the Upgrade Wizard to install upgrade kits after they are retrieved. The process for invoking the Upgrade Wizard varies by subscriber type:

- For client subscribers, the Upgrade Wizard is invoked automatically after an upgrade is retrieved.
- For Siebel Server subscribers, the Upgrade Wizard is invoked manually (see *Retrieving Upgrade Kits for Siebel Regional Node Servers*).

The process for retrieving and installing upgrade kits by Siebel Client subscribers is described in the following topics:

- *Retrieving Optional Upgrade Kits for Developer Web Clients*
- *Retrieving Required Upgrade Kits*

Retrieving Optional Upgrade Kits for Mobile Web Clients

This topic describes how to retrieve and install an optional upgrade kit for a Mobile Web Client. Optional upgrade kits are only for client subscribers. Upgrade kits for server subscribers must always be required.

This topic is part of *Retrieving and Installing Upgrade Kits*.

> **Note:** Siebel Web Clients (the client type that is not installed using a Siebel installer) are not Siebel Anywhere subscribers.

The following procedure describes how to retrieve and install an optional upgrade kit for a Mobile Web Client.

> **Note:** The following procedure restarts the Mobile Web Client computer. It is recommended that users save their work and exit from other applications before starting this procedure.

To retrieve an optional upgrade kit for a Mobile Web Client

1. Invoke the Mobile Web Client.
2. Choose File, then Synchronize, and then Database.
3. In the Siebel Remote dialog box, click Synchronize.

   Synchronization takes place and is complete when the opposing arrows in the corner of the screen disappear.
4. Navigate to the User Preferences screen, then the Component Upgrades view.
5. In the Component Upgrades list, select the desired upgrade component, and check the check box under the Upgrade column.

   A check appears. The Status field value is Upgrade Available for those components where optional upgrade kits are available.
6. Click Upgrade Selected Components to display the Siebel Anywhere Kit Download dialog box, and then complete the following substeps:
   a. Follow the instructions in the Siebel Anywhere Kit Download dialog box to start a Siebel Remote session and retrieve the upgrade kit necessary to upgrade your system.
   b. Click OK to close the Siebel Anywhere Kit Download dialog box.
7. Synchronize *again* to retrieve the corresponding upgrade kits.
8. Click Upgrade Selected Components again to invoke the Upgrade Wizard for the desired upgrade component.

   After the Upgrade Wizard completes, it will automatically restart the Mobile Web Client.
9. Log in, repeat the previous steps and, from the Component Upgrades list, verify the Status is Version OK for the desired upgrade component. The following describes the fields in the Component Upgrades list.

Field Name	Comments
Upgrade	Check this field to upgrade the component.
Component Name	The name of the upgrade component.
Status	The status of the upgrade component. Values can include the following: *Upgrade Available.* An optional upgrade kit is available for this component. *Upgrade Required.* A required upgrade kit is available for this component. *Version OK.* The version of this upgrade component is current. *Upgrade Not Available.* The upgrade component is outdated and must be updated; however, no upgrade kit is available to do this. Contact your Siebel administrator.
Current Version	The version of the component on the client computer.
Minimum Version	The minimum required version of the component.

Field Name	Comments
Maximum Version	The maximum required version of the component.
Upgrade Size (bytes)	The size of the upgrade kit.
Download Time (mins)	The estimated time to download the upgrade kit.

Retrieving Optional Upgrade Kits for Developer Web Clients

Siebel Developer Web Clients can retrieve optional upgrade kits as described in the following procedure.

This topic is part of *Retrieving and Installing Upgrade Kits*.

To retrieve an optional upgrade kit for a Developer Web Client

1. Start the Developer Web Client.
2. Navigate to the User Preferences screen, then the Component Upgrades view.

   In the Component Upgrades list, the upgrade status is Upgrade Available for those components where optional upgrade kits are available. If you do not see the Upgrade Available status for the component you want to upgrade, then contact your Siebel administrator, because this discrepancy might indicate a configuration problem.

   After the administrator corrects the problem, refresh your own view of the Component Upgrades list and reinspect the upgrade status for components that have optional upgrade kits.
3. In the Component Upgrades list, select the desired upgrade component, and select the check box in the Upgrade column for the desired upgrade component.
4. Click Upgrade Selected Components.

   This will download the upgrade kit, shut down the Developer Web Client application, and start the Upgrade Wizard. After the Upgrade Wizard completes, it will automatically restart the Developer Web Client application. For some Third Party Software upgrade kits, the client computer might shut down and restart.
5. Log in and, from the Component Upgrades list, verify the Status is Version OK for the desired upgrade component.

Retrieving Required Upgrade Kits

Both Mobile and Developer Web Clients perform a version check during login to a Siebel application. Mobile Web Clients also perform a version check during synchronization.

This topic is part of *Retrieving and Installing Upgrade Kits*.

If the version check detects that an upgrade is required, then a dialog box appears asking the user whether to upgrade or not. If the user chooses Yes, then the upgrade kit will be downloaded. The following procedures for both the Mobile and Developer Web Clients illustrate this point.

Upgrade kits are not automatically retrieved upon version check. A prompt appears, asking if the user would like to retrieve the upgrade kit. The upgrade kit is downloaded only if the user clicks yes to download the upgrade kit.

The version check verifies the components used by the client configuration; multiple required components can be displayed in the dialog box.

> **Note:** Synchronizing a Mobile Web Client automatically includes copying version information for Siebel Anywhere from the server to the local database. This allows each Mobile Web Client to detect new required and optional upgrades for the client's configuration without waiting for individual database transactions concerning versions to be synchronized.

If the user answers Yes, then every required upgrade kit for the components used by the client configuration is automatically retrieved and the Upgrade Wizard is started.

If the user answers No, then the Developer Web Client starts in read-only mode against the server database, and the Mobile Web Client starts in read-only mode against the local database.

If the user answers No, then the user's session continues in read-only mode against either the local or server database, until the upgrade has been completed. A prompt notifies the user of this status.

While operating in read-only mode, both Developer and Mobile Web Client users can view screens, views and data as usual, but they cannot make changes. Mobile Web Client users can synchronize, but can only download files; they cannot send, receive, or apply database changes. Read-only mode prevents users from corrupting data with an outdated or invalid Siebel application configuration.

Both Developer and Mobile Web Client users exit read-only mode when the Siebel application is restarted following a successful upgrade. At that point, a Mobile Web Client user can perform a full synchronization, sending to the Siebel Server any local database changes that predate the upgrade.

Retrieving Required Upgrade Kits for Mobile Web Clients

If a user starts the Mobile Web Client in read-only mode, then the user is prompted to upgrade at each synchronization session.

The version check for Mobile Web Clients occurs during synchronization. If the version check detects the need for an upgrade, then the user receives a prompt indicating a possible upgrade. The remote user logs in again to synchronize the server and local database.

To retrieve required upgrade kits for Mobile Web Clients

1. Invoke the Mobile Web Client.
2. Choose File, then Synchronize, and then Database.

   A dialog box appears indicating there is a required upgrade available.
3. From the dialog box, click Yes and follow the prompts to invoke the Upgrade Wizard.

   The Upgrade Wizard will shut down the Siebel client, install the upgrade kit, and automatically restart the client unless there is an error while installing the upgrade kit.
4. Log in to the Siebel application.
5. Navigate to the User Preferences screen, then the Component Upgrades view.

6. Verify that the Status and Current Version columns contain the updated information.

Retrieving Required Upgrade Kits for Developer Web Clients

Retrieving Developer Web Clients is similar to that of Mobile Web Clients, except version checking only occurs during login.

To retrieve required upgrade kits for Developer Web Clients

1. Start the Developer Web Client.
   Logging in will trigger a version check and will prompt for retrieval of a required upgrade kit. If you choose not to upgrade at this time, then you can only operate in read-only mode.
2. Click Yes.
   The Siebel client downloads the appropriate upgrade kits.
   After the retrieval is complete, the client invokes the Upgrade Wizard before exiting. After the client exits, the Upgrade Wizard applies the upgrade kits and automatically restarts the client when it is done.
3. Log in to the Siebel application.
4. Navigate to the User Preferences screen, then the Component Upgrades view.
5. In the Component Upgrades list, verify that Status is Version OK.

Starting the Upgrade Wizard

Subscribers use the Upgrade Wizard to install upgrade kits after they are retrieved. This task is a step in *Process of Retrieving, Installing, and Testing Upgrade Kits*.

The process for invoking the Upgrade Wizard varies by subscriber type:

- For Siebel Server subscribers, the Upgrade Wizard is started manually (see *Retrieving Upgrade Kits for Siebel Regional Node Servers*).
- For client subscribers, the Upgrade Wizard is started automatically after an upgrade is retrieved.
  The wizard automatically shuts down the Siebel client. It then displays a progress page as it completes the upgrade kit items for the upgrade kits that were retrieved.

Note: The items displayed on the Upgrade Wizard page vary, depending on the number and type of upgrade kits.

When the Upgrade Wizard has successfully installed the upgrade kit, the Siebel client is automatically restarted.

About Client Upgrade Error Recovery

Before beginning the upgrade process, the Upgrade Wizard makes a backup copy of every file affected by the upgrade. If errors are encountered in the upgrade process, then the Upgrade Wizard stops processing. A message indicates that the Upgrade Wizard did not complete successfully and suggests that you examine the log file for details. It also indicates to restart the Siebel application and attempt the upgrade again.

The next time the user restarts the Siebel client, a prompt appears that states that there had been an upgrade in process.

After the user clicks OK, another prompt appears where the user can cancel the upgrade or restart the upgrade from the point of failure. Canceling the upgrade will result in a cleanup. The user will not be able to restart this upgrade.

The Upgrade Wizard functionality includes error recovery for upgrades on individual files, such as Siebel database schema extensions or client configurations. However, rolling back upgrades always requires manual intervention.

Retrieving Upgrade Kits for Siebel Regional Node Servers

Upgrade kits for Siebel Regional Node Server configurations are retrieved automatically by the Siebel Server. These upgrade kits might include a Siebel database schema or user-defined files to be installed with version tracking. This task is a step in *Process of Retrieving, Installing, and Testing Upgrade Kits*.

For regional databases, the Siebel administrator must take care to distribute the Siebel Database Schema upgrade kit under the correct configuration. The default configuration that ships with Siebel Business Applications is Siebel Regional Server. You can create your own configuration for Regional Node Servers to suit the particular needs of your organization.

For more information about working with the server components used for Siebel Replication Manager, see *Siebel Remote and Replication Manager Administration Guide* and *Siebel System Administration Guide* .

Configuring Siebel Regional Node Servers Using the Server Manager GUI

The following procedure describes how to set the correct configuration for Siebel Regional Node Servers using the Server Manager GUI.

To set the correct configuration for Siebel Regional Node Servers

1. Navigate to the Administration - Server Management screen, then the Servers view.
2. From the Siebel Servers list, select the appropriate server, and click on the Server Parameters tab.

   The Server Parameters list appears.
3. Modify the parameters as necessary.

   Modify the server parameter Upgrade Component to your configuration name. This configuration name can be Siebel Regional Server or the name of your custom configuration.

Configuring Siebel Regional Node Servers Using the Server Manager Command Line

The following procedure describes how to set the correct configuration for Siebel Regional Node Servers using the Server Manager command line.

To set the correct configuration for Siebel Regional Node Servers using command line

- From the srvrmgr command line, enter:

```
srvrmgr.exe /g gateway_srvr_host /e enterprise /s server_name /u user_name /p password
change param UpgComponent="Siebel Regional Server" or "name of your custom configuration"
```

If the version check recognizes a required upgrade, then the Siebel Server fails to start and logs an error message in the siebsrvr.log or repagent_$*taskid*.log file similar to the following:

```
Please contact HQ administrator. An upgrade may be required.
```

With the Regional Node Server and all tasks stopped, the Siebel administrator must manually invoke the Upgrade Wizard.

Manually Invoking the Upgrade Wizard on Windows

The following procedure describes how to manually invoke the Upgrade Wizard on Microsoft Windows.

To manually invoke the Upgrade Wizard on Windows

1. Open a command prompt window on the Windows server where the Regional Node Server is installed.
2. Navigate to the SIEBSRVR_ROOT `\BIN` directory, where SIEBSRVR_ROOT is the root directory of the Regional Node Server installation.
3. Copy the file siebupg.exe to another directory, and navigate to that directory.
4. Execute the copy of siebupg.exe to start the Upgrade Wizard.

   Follow the prompts on the wizard pages.
5. Restart the Regional Node Server.
6. Repeat the previous steps on any other Regional Node Server, as appropriate.

Manually Invoking the Upgrade Wizard on UNIX

The following procedure describes how to manually invoke the Upgrade Wizard on UNIX platforms.

To manually invoke the Upgrade Wizard on UNIX

1. Open a shell on the UNIX server where the Regional Node Server is installed.
2. Navigate to the SIEBSRVR_ROOT/`BIN` directory, where SIEBSRVR_ROOT is the root directory of the Regional Node Server installation.
3. Execute the file srvrupgwiz to start the Upgrade Wizard.

   Follow the prompts on the wizard pages.
4. Restart the Regional Node Server.
5. Repeat the previous steps on any other Regional Node Server, as appropriate.

If the Upgrade Wizard fails, then an error message is logged. Exit the Upgrade Wizard, correct the condition that caused the error, and restart the Upgrade Wizard using the *copy* of the siebupg.exe file. Upon restarting, Upgrade Wizard prompts you either to cancel the upgrade, restoring the Regional Node Server to its preupgrade state (if possible), or

to retry the upgrade from the point of failure. You must restart the Regional Node Server to initiate a version check to restart the upgrade process. To restart, choose Retry.

The Regional Node Server will be down until you successfully recover from the failed upgrade. This downtime can affect many users. After a successful installation of an upgrade, it will take manual intervention to roll it back.

Testing Upgrade Kits

This topic describes how to test upgrade kits. This task is a step in *Process of Retrieving, Installing, and Testing Upgrade Kits*.

The recommended method for testing upgrade kits is to create and distribute an upgrade kit to a limited and controlled number of users. In this method, you can assign specific testing users to the Siebel Test Client configuration so your regular subscribers will not be allowed to install any upgrade kits in the testing phase. Distribute the upgrade kit only to the Siebel Test Client configuration. See *Using the Siebel Test Client Configuration for Testing*.

> **CAUTION:** When testing new upgrade kits, it is strongly recommended that you perform tests using both Developer and Mobile Web Clients. If for any reason Mobile Web Client users experience problems installing or downloading the new upgrade kit, then the problems will be significant.

After an upgrade kit has been thoroughly tested under the Siebel Test Client configuration, you can then distribute it to the appropriate production configurations.

To optimize the setup process for new users, it is recommended that you create a master installation image for each configuration your Siebel Anywhere subscribers use, and keep these images up to date. If all current upgrade kits are installed to create a current image, then you can use the image to set up new subscribers, so each new subscriber will not need to download and install many upgrade kits. This method also makes it safer to deactivate old upgrade kits, since new subscribers will receive the necessary software by means of installation images, rather than being required to install old upgrade kits.

> **Note:** Siebel Anywhere does not deliver Siebel Database Schema upgrade kits to individual Developer Web Client users, because they connect to shared databases. Test database extensions in the development environment.

Using the Siebel Test Client Configuration for Testing

Siebel Anywhere provides a special configuration called Siebel Test Client to serve as your test user. By default, the Siebel Test Client configuration is representative of the other subscriber configurations initially provided, but it can be modified to reflect your active configuration.

To test an upgrade kit

1. Create a user and assign this user with the Siebel Test Client configuration in one of the following ways:
   - Assign the user to the configuration, as described in *Assigning Employees to a Configuration*.
   - Install a client and set ComponentName to *Siebel Test Client* in the active CFG file.
2. Define, activate and apply the upgrade kit as an optional upgrade kit and distribute to the appropriate configuration.
3. Verify that the Siebel Test Client configuration contains the components being upgraded with the upgrade kit.

4. Notify the selected users assigned to the Siebel Test Client configuration to log in. Instruct them to use the procedures described in *Retrieving and Installing Upgrade Kits* to retrieve and install the upgrade kit.
5. After test users install the upgrade kit, have them test it by running the application that was upgraded.

Have all test users report any unexpected behavior in the application. Give special attention to areas of the application that changed due to the upgrade.

If the test is unsuccessful, then see *Troubleshooting for Siebel Anywhere* for possible solutions.

> **Note:** If you want to change your optional upgrade kit to be required after testing, then see *Converting an Optional Upgrade Kit to a Required Upgrade Kit*.

Uninstalling a Delta Patch

Occasionally, as part of testing, troubleshooting, or as preparation before installing a new release, you might want to uninstall the client files associated with a delta patch.

> **Note:** As an alternative to uninstalling a delta patch, you can use the uninstallation methods described in the *Siebel Installation Guide* or in the documentation for the Siebel Patchset release.

Uninstalling a Delta Patch but Retaining Previous Releases

As part of testing or troubleshooting, sometimes you might want to uninstall a single Siebel Innovation Pack release or Siebel Patchset release, while leaving the prior installed release or releases intact. For example, you might want to remove version 17.0, but retain version 8.1.1 and version 8.1.1.7, or retain version 8.2.2 and version 8.2.2.2. You can use the procedure in this topic to uninstall a corresponding delta patch.

> **Note:** Using this method, you can uninstall only a delta patch, not any other type of patch release. A *delta patch* is software that was installed using a Siebel Anywhere upgrade kit of type Siebel Client Executables with Select Type of Installation set to Delta Install. For more information, see *Roadmap for Creating a Delta Installation Siebel Client Executables Upgrade Kit*.

The following procedure describes how to uninstall a delta patch.

To remove a delta patch but retain the prior installed release or releases

1. Navigate to the existing root directory of the client installation (SIEBEL_CLIENT_INSTALL), such as `c:\siebel\client`.
2. Locate the following directories, which were created when the delta patch was installed:
   - `PATCHBACKUP_BASE`. This folder is for the base portion of the installation that was performed through the delta patch.
   - `PATCHBACKUP`_language-code, where *language-code* is the code for each language that is used in the client installation. For example, the directory might be called `PATCHBACKUP_ENU` for U.S. English. This folder is for a language-specific portion of the installation that was performed through the delta patch.
   - `PATCHBACKUP`_ language-code, where *language-code* is the code for each language that is used in the client installation. SIA applies for all Siebel Industry Applications. For example, the directory might be called

PATCHBACKUP_ENUSIA for U.S. English. This folder is for a language-specific portion of the installation that was performed through the delta patch.

- PATCHBACKUP_. This folder is for the base portion of the installation, plus all language-specific portions of the installation, that was performed through the delta patch.

3. In each of the directories that you located rename the file *unpatch.bak* to *unpatch.bat*.
4. In each of these directories, execute *unpatch.bat*.

Note: The unpatch.bat files can be executed from the various directories in any order.

6 Activating, Applying, and Distributing Upgrade

Activating, Applying, and Distributing Upgrade Kits

This chapter discusses activating, applying, and distributing upgrade kits, plus related tasks. It includes the following topics:

- *Process of Completing Upgrade Kit Creation*
- *Activating an Upgrade Kit*
- *Controlling the Order of Upgrade Kit Installation*
- *Deactivating an Upgrade Kit*
- *Applying an Upgrade Kit*
- *Converting an Optional Upgrade Kit to a Required Upgrade Kit*
- *Distributing Upgrade Kits*
- *Disabling Siebel Anywhere*

Process of Completing Upgrade Kit Creation

The remaining three steps in the process of creating an upgrade kit are activating, applying, and distributing the upgrade kit.

To complete upgrade kit creation, perform the following tasks:

1. *Activating an Upgrade Kit*. This step gathers all the files for the upgrade and creates a single compressed archive file.
2. *Applying an Upgrade Kit*. This step updates the compiled information string in the database with the new version information. If you have not already done so, then, during this step, you indicate whether or not an upgrade kit will be required.
3. *Distributing Upgrade Kits*. This step makes the upgrade kit available to designated subscribers as either a required or an optional upgrade kit, depending on your selection in Step *Process of Completing Upgrade Kit Creation* of this process.

> **Note:** There are additional instructions for distributing Siebel Database Schema upgrade kits. For information, see *Process of Updating the Siebel Database Schema*.

Activating an Upgrade Kit

After you define the upgrade kits for an upgrade, you must activate each upgrade kit. The status of an upgrade kit must be Pending before activating it. Activating an upgrade kit gathers the files to be included in the upgrade kit and compresses them into a single archive on the Siebel File System. It also sets the status of the upgrade kit to Active.

This task is a step in *Process of Completing Upgrade Kit Creation*

To activate an upgrade kit

1. Navigate to the Administration - Siebel Anywhere screen, then the Upgrade Kits view.

   The Upgrade Kits view appears.
2. Select the upgrade kit you want to activate.

   The Status field must show that it is Pending.
3. Click Activate.

   A dialog box appears indicating activation is in progress.

   If you are activating a large upgrade kit, then your browser might be unresponsive for a few minutes after you click Activate. The delay in response occurs because the client application must wait for a reply from the server before proceeding. During the wait, no hourglass is displayed. When the client application receives the server's reply, the Status field changes to Active.
4. Repeat the previous steps to activate each upgrade kit required for your upgrade.

> **Note:** Activating the upgrade kit does not affect the version information stored in the database. Activating an upgrade kit populates the File Size field for that upgrade kit. The upgrade kit is not available to subscribers until it has been applied and distributed.

When you activate the upgrade kit, the upgrade kit items you see in this view turn into item entries in the upgrade.ucf file, which is part of every Siebel Anywhere upgrade kit.

Controlling the Order of Upgrade Kit Installation

Usually, when you distribute upgrades to a configuration requiring several related components, the order in which upgrade kits are installed is undefined. If you must control the installation order of upgrade kits, then you can do this by making one upgrade kit dependent upon another. If this is the case, then do not use the Upgrade Kit Wizard to activate the upgrade kit.

An example of this type of situation is when you have one Siebel Repository Export Import upgrade kit (Upgrade Kit 1) and another Siebel Client Customer Revisions upgrade kit (Upgrade Kit 2), and you need Upgrade Kit 1 to be installed first.

To set the order of upgrade kit installation

1. Navigate to the Administration - Siebel Anywhere screen, then the Upgrade Kits view.

The Upgrade Kits list appears.

2. In the Upgrade Kits list, select the upgrade kit that you need installed second (for example, Upgrade Kit 2, the Siebel Client Customer Revisions upgrade kit).
3. If the upgrade kit is already activated, then perform one of the following actions:
   - Manually set the status to Pending.
   - Deactivate the upgrade kit, which changes the status to Pending.
4. In the Upgrade Kit Component list (lower list in the same view), make sure that the upgrade kit to be installed second is still selected in the Upgrade Kits list, and then click New.

   The Upgrade Components dialog box appears.
5. Select the component Upgrade Kit 1 upgrades; for example, Siebel Client Repository__*language-code*.
6. In the new record, set the Min Old Version to the New Version of the upgrade kit (the Siebel Repository Export Import upgrade kit).
7. Click to activate the Upgrade Kits applet.
8. Select the upgrade kit that you need installed second again and click Activate.

This procedure forces first the installation of Upgrade Kit 1 (Siebel Repository Export Import upgrade kit), followed by Upgrade Kit 2 (Siebel Client Customer Revisions upgrade kit). This interlinked series can be extended to three or more upgrade kits.

Deactivating an Upgrade Kit

Sometimes you might need to deactivate an active upgrade kit. For example, perhaps there was a last-minute decision to delete an upgrade kit. (You must first deactivate an upgrade kit before deleting it.)

Observe the following guidelines when deciding whether to deactivate an upgrade kit:

- If there was a problem with the upgrade kit, then deactivate the upgrade kit and create a replacement upgrade kit to upgrade the users who had problems with the initial upgrade kit.
- If later upgrade kits depend on the upgrade kit you are considering for deactivation, then perform the following actions to make sure that users will be able to download and install the later upgrade kits:
  - Make sure that every user who needs the upgrade kit that will be deactivated has already installed the upgrade kit. This includes any infrequent users who might log in too infrequently to have been prompted to install the upgrade kit.

    > **CAUTION:** Deactivating a previously distributed upgrade kit could cause mobile users to have an unsuccessful synchronization and place those users unnecessarily into a read-only state.

  - Make sure that you can clone new client installations from existing up-to-date clients. This method is the preferred method for providing new users with material that was formerly delivered in one or more upgrade kits. New users cannot download later upgrade kits if a prerequisite upgrade kit has been deactivated.

To deactivate an upgrade kit

1. Navigate to the Administration - Siebel Anywhere screen, then the Upgrade Kits view.

   The Upgrade Kits view appears.

2. Select the upgrade kit that you want to deactivate.

   The Status field indicates Active.
3. Click the menu button and select Deactivate.

   A warning message asks you to confirm. If appropriate, then click OK.

   The Status field now indicates Pending.
4. Repeat the previous steps for each upgrade kit that you want to deactivate.

Applying an Upgrade Kit

After you activate the upgrade kits for an upgrade, you must apply each upgrade kit. Applying an upgrade kit updates the compiled information string in the database with the component version information. This is an essential step in creating an upgrade kit. This is also the step in which you indicate whether an upgrade kit is a required upgrade kit.

This task is a step in *Process of Completing Upgrade Kit Creation*

Upgrade.ucf is the driver file for the Upgrade Wizard to apply the upgrade kit. It contains an ordered list of action items for the Upgrade Wizard to execute during the installation of the upgrade kit.

The following procedure describes how to apply an upgrade kit. This updates the compiled information string in the database with the component version information.

> **Note:** If a replacement upgrade kit uses the same values for New Version, Minimum Old Version, and Maximum Old Version as the deactivated upgrade kit that it replaces, and if the deactivated upgrade kit was previously distributed, then you do not need to apply or distribute the replacement upgrade kit.

To apply an upgrade kit

1. Navigate to the Administration - Siebel Anywhere screen, and, from the link bar, click Upgrade Kits.

   The Upgrade Kits view appears.
2. Select the upgrade kit that you want to apply.

   The Status field must show that it is Active.
3. Click Apply Versions.

   The Apply Upgrade Kit Version Information dialog box appears. Review the information in this dialog box:
   - The Min Version and Max Version numbers in this dialog box apply to the version of the component that can be used to bring up the application in read and write mode. If users have a version below the

minimum and choose not to install the upgrade, then they can only access the application in a read-only mode.

- If you click OK without clicking Require Upgrade Kit, then you are making it an optional upgrade kit (that is, after the upgrade kit is distributed, your subscribers can use Siebel Business Applications without upgrading, if their version is between the minimum and maximum).
- If you click Require Upgrade Kit, then the minimum version changes to match the maximum version, which is the new version. In this case, the subscribers must upgrade after the configuration is distributed; otherwise, they can only start the application in read-only mode.

> **Note:** It is strongly recommended that you create upgrade kits as optional and then test to make sure that the upgrade kit is functioning properly. After you thoroughly test an upgrade kit using retrieval and installation, you can return to the Upgrade Kits view and reapply and distribute the upgrade kit as a required upgrade kit. See *Distributing Upgrade Kits* for more information.

4. Click OK if the version information is correct or click Cancel to exit the dialog box without applying the upgrade kit.

   If you click OK, then a prompt reminds you that you must distribute the upgrade kit to make it available to subscribers.
5. Repeat the previous steps for each upgrade kit in this upgrade.

   Continue to *Distributing Upgrade Kits* for distributing updated version information for upgrade kits.

> **Note:** If there are multiple upgrade kits of the same type that are Siebel Database Schema or Siebel Client Executables, then it is strongly recommended you deactivate the old upgrade kits. This will save time for your users by preventing the downloading of outdated upgrade kits. (The version information for a deactivated upgrade kit can be retrieved, but a deactivated upgrade kit, itself, cannot be retrieved. When a user tries and fails to retrieve a deactivated upgrade kit, an automatic attempt is made to retrieve a later upgrade kit that is currently activated.)

Converting an Optional Upgrade Kit to a Required Upgrade Kit

After thoroughly testing an optional upgrade kit, you can reapply the upgrade kit to convert it to a required upgrade kit

To convert an optional upgrade kit to a required upgrade kit

1. Navigate to the Administration - Siebel Anywhere screen, then the Upgrade Kits view.

   The Upgrade Kits view appears.
2. From the Upgrade Kits list, select the optional upgrade kit that you want to convert.

   The Status field indicates Active.
3. Click Apply Versions.

   The Apply Upgrade Kit Version Information dialog box appears. Review the information in this dialog box.
4. Click Require Upgrade Kit to make the upgrade kit required.

   The Minimum and Maximum Version numbers are now the same because the upgrade kit is now required.

5. Click OK if the version information is correct or click Cancel to exit the dialog box without applying the upgrade kit.

   If you click OK, then a prompt reminds you that you must distribute the upgrade kit to make it available to subscribers. See *Distributing Upgrade Kits*.

Distributing Upgrade Kits

After applying an upgrade kit, you must distribute it to one or more configurations. For general information about configurations, see *Upgrade Configurations*.

This task is a step in *Process of Completing Upgrade Kit Creation*

> **Note:** If a replacement upgrade kit uses the same values for New Version, Minimum Old Version, and Maximum Old Version as the deactivated upgrade kit it replaces, and if the deactivated upgrade kit was previously distributed, then you do not need to distribute the replacement upgrade kit.

Upgrade kits are distributed to one configuration at a time. The distribution process makes the upgrade kit's component version information available to all the subscribers who are associated with that configuration, giving the subscribers their first opportunity to download and install the upgrade kit. Neither required nor optional upgrade kits are available to subscribers until the version information is distributed.

The version information that is made available during the distribution process includes version information for all the related components of the configuration, not just the component that the new upgrade kit is designed to upgrade. Technically, therefore, when you distribute a new upgrade kit to a configuration, you make that upgrade kit's version information available to the configuration, and you redistribute the version information of other active upgrade kits for the configuration's related components. In practice, however, you generally can proceed as if you were distributing only the one upgrade kit.

After distribution, if the upgrade kit is required, then subscribers are automatically prompted to retrieve and install it. If the upgrade kit is optional, then subscribers use the Component Upgrades view and Upgrade Wizard to locate and install the upgrade kit.

> **CAUTION:** Before you distribute an upgrade kit, you must define, activate, and apply it. Before you distribute an upgrade kit to any configuration containing production users, it is strongly recommended that you use the Siebel Test Client configuration to test the upgrade kit thoroughly. It is particularly important to test any required upgrade kit before distributing for general use, because Mobile Web Client users can suffer unnecessary delays while downloading a defective required upgrade kit, and they can lose read and write access to Siebel Business Applications until a correctly functioning upgrade kit is available.

The following procedure describes how to distribute an upgrade kit from the Upgrade Configurations view. The same procedure can also be performed from the Employees view.

> **Note:** There are additional instructions for distributing Siebel Database Schema upgrade kits. For information, see *Process of Updating the Siebel Database Schema*.

To distribute an upgrade kit

1. Navigate to the Administration - Siebel Anywhere screen, then the Upgrade Configurations view.
2. In the Upgrade Configurations view, select the appropriate Anywhere subscriber configuration.
3. Inspect the Upgrade Components list to verify that the components in the upgrade kit are related components for the selected configuration.

   For example, for a Siebel Innovation Pack release, the related components must include the Siebel Client Executables component for the base upgrade kit.
4. Click Distribute.

   This action makes the version information in the database available to subscribers with that particular configuration.

   A dialog box appears to confirm the intent to distribute the related components to the configuration.
5. Repeat the previous steps for other configurations, as applicable.

For more information about special cases of distribution, see:

- *Limiting the Distribution of an Upgrade Kit*
- *Distributing Different CFG Files to Different Users*

Limiting the Distribution of an Upgrade Kit

You can limit the distribution of a upgrade kit to certain subscribers by temporarily assigning subscribers to a new configuration. For example, you might have a situation where you want to send a set of reports only to management, or you want to limit the users for testing an upgrade kit.

This topic is part of *Distributing Upgrade Kits*.

To limit upgrade kit distribution for a new configuration

1. Define the upgrade kit that you will be distributing.

   For more information, see *Defining Upgrade Kits*
2. Activate the upgrade kit that you will be distributing.

   For more information, see *Activating an Upgrade Kit*.
3. Apply the upgrade kit that you will be distributing.

   For more information, see *Applying an Upgrade Kit*.
4. Either create a new configuration that you will use to distribute the upgrade kit to specific subscribers, or use the Siebel Test Client configuration for this purpose.

   For more information, see *Creating a New Configuration*.
5. Associate the configuration with one or more components that correspond to the upgrade kit or upgrade kits that you want to distribute.

   For more information, see *Adding Components to a Configuration*.
6. Add the logins for the subset of users that you want to receive this upgrade kit to the Employee view for this configuration.

   For more information, see *About Assigning Employees to a Configuration*.

7. Distribute active upgrade kits to the configuration.
   For more information, see *Distributing Upgrade Kits*.
   When you select a configuration and click Distribute, the subscribers associated with the configuration gain access to version information for all of the related components for the configuration. This gives those subscribers potential access to all active upgrade kits for those components, including any new upgrade kits that have been defined, activated, and applied. Subscribers associated with other configurations do not gain access to your new upgrade kit until you distribute to their configurations.
8. Remove subscribers from the configuration.
   Give the subscribers who are associated with the configuration enough time to download the upgrade kit. However, you must disassociate these subscribers from this special configuration as soon as possible. Ending a user's dynamic configuration assignment returns that user to the original configuration assignment defined in that user's CFG file. Upgrade kits targeted for the original configuration are again available to the user.
   For more information, see *Removing Employees from a Configuration*.

Distributing Different CFG Files to Different Users

It is possible to distribute different CFG files to different sets of users. For example, you might have two groups of users, Sales Group users and Engineering Group users, where each group uses a different CFG file with their respective Siebel applications.

This topic is part of *Distributing Upgrade Kits*.

Distributing CFG Files to Different Groups of Users

The following procedure describes how to use Siebel Anywhere to distribute a new CFG file to each group of users.

To distribute different CFG files to different groups of users

1. Prepare all of the CFG files your various groups will use.
2. Start the Siebel application.
3. Navigate to the Administration - Siebel Anywhere screen, then the Upgrade Component List view.
   The Upgrade Components list appears.
4. Create a component for each applicable language, such as Sales Group CFG__ENU, that is similar to one of the default components for language-specific CFG files. Use the Copy Record option and set the values for the columns for each record as shown in the following tables.
   Example values for Record 1:

Field Name	Value
Name	Sales Group CFG__ENU
Component Type	Siebel CFG File
Min Version	100

Field Name	Value
Max Version	Leave blank for a new component
Locate Method	CFG
Locate Information	Siebel,ClientRootDir,BIN\enu\SalesGrp.cfg
Version Method	CFG
Version Information	Siebel,Version
Component Type	Sales Group CFG__ENU

Example values for Record 2:

Field Name	Value
Name	Engineering Group CFG__ENU
Component Type	Siebel CFG File
Min Version	100
Max Version	Leave blank for a new component
Locate Method	CFG
Locate Information	Siebel,ClientRootDir,BIN\enu\EngGrp.cfg
Version Method	CFG
Version Information	Siebel,Version

5. Create a configuration for each language combination for each group of users (such as Sales Group and Engineering Group), and add the appropriate CFG components you just created as related components.

   See *Planning and Preparing to Use Siebel Anywhere* for details.

6. Navigate to the Administration - Siebel Anywhere screen, then the Upgrade Kits view.
7. Use Auto Create to create one upgrade kit for each group of users.

   For example, you might create one upgrade kit for Sales Group users, using the Sales Group CFG__*language-code* component and SalesGrp.cfg file, and another for Engineering Group users, using the Engineering Group CFG__*language-code* component and EngGrp.cfg file.
8. Apply the upgrade kits, and click on the Require Upgrade Kit button in the Apply Upgrade Kit Version Information dialog box.

The following procedure describes how to test the distribution of different CFG files to different groups of users.

Testing the Distribution of CFG Files to Different Groups of Users

The following procedure describes how to test the distribution of different CFG files to different groups of users.

To test distributing CFG files to different groups of users

1. Navigate to the Administration - Siebel Anywhere screen, then the Upgrade Configurations view.
2. Add the new CFG components (such as Sales Group CFG__ENU and Engineering Group CFG__ENU) to the Siebel Test Client configuration.
3. Add two or more users to the Employees list for Siebel Test Client.
4. Distribute the Siebel Test Client to verify that you can retrieve the kit successfully.
5. Make sure that your test users' CFG files reflect the correct ComponentName in their CFG files.

   For example, Sales Group users must have this parameter set to Sales Group Configuration while Engineering Group users must have this parameter set to Engineering Group Configuration. Also, have these users verify that an exact match exists between the ComponentName in the CFG file and the configuration name in the Administration - Siebel Anywhere screen, Upgrade Configurations view.
6. Distribute the final configurations (such as Sales Group Configuration and Engineering Group Configuration) to send the kit to all applicable users.

Disabling Siebel Anywhere

Sometimes you might not want your users to be prompted to install upgrade kits when they log in to Siebel applications. The following procedures describe how to disable Siebel Anywhere version checking and the ability to install Siebel Anywhere upgrade kits for Developer Web Clients and Mobile Web Clients.

Disabling Siebel Anywhere for Siebel Developer Web Clients

The following procedure describes how to disable Siebel Anywhere for Siebel Developer Web Clients.

To disable Siebel Anywhere for Developer Web Clients

1. Navigate to the Administration - Siebel Anywhere screen, then the Upgrade Configurations view.

2. Use one of the following methods to disable Siebel Anywhere version checking and related user prompts:
   - Rename the appropriate configurations. For example, rename the Siebel Sales Client as Old Siebel Sales Client, and so on. Then, redistribute current version information to the renamed configuration.
   - Delete all of the related components in a configuration and redistribute current version information to that configuration.

Disabling Siebel Anywhere for Siebel Mobile Web Clients

The following procedure describes how to disable Siebel Anywhere for Siebel Mobile Web Clients.

To disable Siebel Anywhere for Mobile Web Clients

1. Follow the instructions in *Disabling Siebel Anywhere for Siebel Developer Web Clients*.
2. In addition, use one of the following methods to disable Siebel Anywhere version checking and related user prompts:
   - Change ComponentName parameter in the CFG file to None.
   - Reextract local database.

   If Mobile Web Clients are associated with a particular configuration through the Administration - Siebel Anywhere screen, Upgrade Configurations, Employees view, then reextracting is the only choice.

7 Performing Additional Siebel Anywhere Tasks

Performing Additional Siebel Anywhere Tasks

This chapter provides supplementary information for specific types of upgrade kits and other special circumstances in which Siebel Anywhere might be used. It includes the following topics:

- *Distributing Additional Language Packs*
- *Example of Global Deployment with Siebel Anywhere*
- *Example of Creating Multiple Upgrade Kits of the Same Type*
- *Process of Updating the Siebel Database Schema*
- *Deleting Old Upgrade Kits*
- *Testing the Consistency of Upgrade Kit Components*
- *Creating Upgrade Kits That Check DLL File Versions*
- *Example of Constructing a Third Party Software Upgrade Kit*

Distributing Additional Language Packs

When you must deliver additional language packs to Mobile or Developer Web Clients, follow these steps:

- (Administrator) Install the new language packs and test them thoroughly.
- For each language pack, create Siebel Client Executables__*language-code* upgrade kits using the package created in the previous step. Use the Delta Install type.
- Distribute the upgrade kits to the appropriate configurations. You must have separate configurations for each combination of languages for your users.

Example of Global Deployment with Siebel Anywhere

This topic provides an example of a global deployment using Siebel Anywhere. The objective is to allow users to access and use the same application in different languages.

You have upgrade components for the languages you installed with your database server. You will map these to the applicable configurations. These configurations will be based upon the languages that different sets of users require to access the same application.

Assumptions for Global Deployment Example

This example makes the following assumptions:

- You have five sets of users for five different languages.
- You will base your application on Siebel Call Center.
- You will have customer revisions, depending on the language.
- You will use the same database.
- Each user only uses the application in one language.

Requirements for Global Deployment Example

You will need:

- Five components for the CFG files
- Five components for Siebel Client Customer Revisions
- Five configurations to distribute the upgrade kits
- One component for Siebel Client Executables (base), including all applicable languages

Global Deployment Example Records

For the records available, use values similar to those in the tables that follow.

Note: Substitute applicable language codes for ENU in these example records.

The following shows example values for defining a Siebel CFG File component.

Field Name	Value
Name	Siebel Call Center CFG__ENU
Component Type	Siebel CFG File
Min Version	100
Max Version	Leave this field blank
Locate Method	CFG

Field Name	Value
Locate Information	Siebel,ClientRootDir,bin\enu\uagent.cfg
Version Method	CFG
Version Information	Siebel,Version

The following shows example values for defining a Siebel Client Customer Revisions component.

Field Name	Value
Name	Siebel Client Customer Revisions Siebel Client Customer Revisions__ENU
Component Type	Customer Revision
Min Version	Leave this field blank
Max Version	Leave this field blank
Locate Method	CFG
Locate Information	Siebel,ClientRootDir,bin\clntrev.cfg Siebel,ClientRootDir,bin\enu\clntenu.cfg
Version Method	CFG
Version Information	Siebel,Version

The following shows example values for defining a Siebel Client Configuration component.

Field Name	Value / Comments
Name	Siebel Call Center ENU
Configuration Type	Siebel Client Configuration

Field Name	Value / Comments
In the related components applet, make sure that the following components are included (assuming that you want to distribute all of these components):	In the related components applet, make sure that the following components are included (assuming that you want to distribute all of these components):
Siebel Call Center CFG__ENU	Required
Siebel Client Customer Revisions	Required
Siebel Client Customer Revisions__ENU	Required
Siebel Client Executables	Required
Siebel Quick Fix	Required
Siebel Client Executables__ENU	Required for language-specific files using delta patch upgrade kits
Siebel Database Schema	Required
Siebel Upgrade Wizard	Required
Third Party - File Attachment	Not required

Note: If you will be creating and distributing language-specific files using delta patch upgrade kits, then you must create a configuration for each language combination. For example, if a user installed both U.S. English (ENU) and French (FRA) on a system, then create a configuration with the components related to these two languages. For more information about multilingual deployments, see *Siebel Global Deployment Guide* .

Example of Creating Multiple Upgrade Kits of the Same Type

If you must distribute multiple custom files to different directories, then create an upgrade kit for each destination directory. Set the values of the Minimum Old Version and Maximum Old Version settings of the upgrade kits so that they create an unbroken sequence.

The following example uses the Customer Revision upgrade kit type (component type). The same logic can be used for other types of upgrade kits when you must distribute multiple upgrade kits of the same type.

Create the first upgrade kit (Upgrade Kit 1) in the usual manner. For example, set Minimum Old Version and Maximum Old Version to the null value, and set New Version to 1.

For the next upgrade kit, set Minimum Old Version to 1, Maximum Old Version to 1, New Version to 2, and so on.

If you have to skip a version number for some reason, then set the Minimum Old Version and Maximum Old Version values appropriately. Always set the value of Minimum Old Version to match the value of New Version for the previous active upgrade kit.

In this example, you need three upgrade kits, but Upgrade Kit 2 was skipped and replaced, because it contained one wrong file. The version information would be as shown in the following table.

Upgrade Kit	Minimum Old Version	Maximum Old Version	New Version
Upgrade Kit 1	Null	Null	1
Upgrade Kit 2 (This upgrade kit contains one wrong file, was never activated, and should be skipped.)	1	1	2
Upgrade Kit 3 (This upgrade kit replaces Upgrade Kit 2.)	1	2	3
Upgrade Kit 4	3	3	4

For more information about the version numbers used by Siebel Anywhere, see *About Siebel Anywhere Version Settings*.

Process of Updating the Siebel Database Schema

This topic describes the overall process of using Siebel Anywhere to update the Siebel database schema. Typical schema changes include:

- Adding extension columns to Siebel base tables
- Adding extension tables to the Siebel database
- Creating or modifying indexes

To update the Siebel database schema, perform the following tasks:

1. Review the information presented in *Review What Happens During a Siebel Database Schema Upgrade*.
2. Make and test the Siebel database schema changes in the development environment, using Siebel Tools. For more information about this step, see *Changing the Siebel Database Schema in a Development Environment*.
3. Move the schema changes to the test environment, using the Siebel Database Configuration Wizard. For more information about this step, see *Moving Siebel Database Schema Changes Between Environments*.
4. Create the necessary upgrade kits in the test environment. For more information about this step, see *Creating Upgrade Kits for Siebel Database Schema Changes*.
5. Prepare the test Mobile Web Clients and Regional Node Servers for the schema changes. For more information about this step, see the following topics:

- *Preparing Mobile Web Clients for Siebel Database Schema Changes*
- *Preparing Siebel Regional Node Servers for Siebel Database Schema Changes*
- *Checking Siebel Regional Node Server Parameters Before a Siebel Database Schema Update*

6. Install the schema changes on the test Mobile Web Clients and Regional Node Servers. For more information about this step, see the following topics:
   - *Installing Siebel Database Schema Changes on Mobile Web Clients*
   - *Installing Siebel Database Schema Changes on Siebel Regional Node Servers*
7. Move the schema changes to the production environment, using the Siebel Database Configuration Wizard. For more information about this step, see *Moving Siebel Database Schema Changes Between Environments*.
8. Create the necessary upgrade kits in the production environment. For more information about this step, see *Creating Upgrade Kits for Siebel Database Schema Changes*.
9. Prepare the production Mobile Web Clients and Regional Node Servers for the schema changes. For more information about this step, see the following topics:
   - *Preparing Mobile Web Clients for Siebel Database Schema Changes*
   - *Preparing Siebel Regional Node Servers for Siebel Database Schema Changes*
   - *Checking Siebel Regional Node Server Parameters Before a Siebel Database Schema Update*
10. Install the schema changes on the production Mobile Web Clients and Regional Node Servers. For more information about this step, see the following topics:
    - *Installing Siebel Database Schema Changes on Mobile Web Clients*
    - *Installing Siebel Database Schema Changes on Siebel Regional Node Servers*

Review What Happens During a Siebel Database Schema Upgrade

This topic provides background information about what happens during a Siebel database schema upgrade.

This task is a step in *Process of Updating the Siebel Database Schema*.

During Siebel database schema upgrades, new and changed objects are identified by comparing the current physical schema with the new virtual schema. Based on this comparison, Upgrade Wizard adds new objects and alters existing objects by using the following:

- To create the new objects: create table, create index, and so on.
- To alter the existing objects: alter table, alter index, and so on.

> **CAUTION:** Downsizing existing columns is not recommended. Do not drop columns or tables. These operations can cause transactions that were created before the Siebel database schema change to be rejected, because data that was included in the earlier transactions might not fit into the new schema.

Normally, the database does not allow downsizing a column that already contains data that could not fit into a new smaller column. If you downsize a column, then the column representation in the repository would be different from the physical schema. Under these conditions, Siebel Remote would fail, among other possible errors.

For example, after downsizing existing columns, one of the columns in the new database is shorter (it is created from the representation in the repository). Then, a local database is reextracted (this includes a new schema). After the Remote user synchronizes, initialization occurs. Upgrade Wizard first applies the new local database (including a new

schema) to a Remote user's computer. Next, Upgrade Wizard applies any server transactions downloaded during synchronization.

If these transactions were generated before the Siebel database schema changed, then Upgrade Wizard tries to insert data that does not fit into the new column, causing the transaction to be rejected. Furthermore, if this occurs on a regional node, then the error might cause an outage.

Changing the Siebel Database Schema in a Development Environment

The first step in deploying Siebel database schema changes is to make the schema changes in a development environment, using Siebel Tools. For more information about this step, see *Using Siebel Tools* and *Configuring Siebel Business Applications* .

This task is a step in *Process of Updating the Siebel Database Schema*.

Keep track of whether your changes include creation or modification of views or applets. This information will affect later steps in the deployment.

Test the changes in the development environment before proceeding to the next part of the process.

Moving Siebel Database Schema Changes Between Environments

When you are ready to move Siebel database schema changes from a developmental environment to a test environment, or from a test environment to a production environment, do not use Siebel Tools to make the changes. Instead, use the Migrate Repository option in the Siebel Database Configuration Wizard to migrate the entire Siebel Repository containing the schema changes to the new environment. For more information about this task, see *Using Siebel Tools* .

This task is a step in *Process of Updating the Siebel Database Schema*.

Test the changes in the new environment before proceeding to the next part of the process.

Creating Upgrade Kits for Siebel Database Schema Changes

The creation of a Siebel Database Schema upgrade kit has two consequences beyond the creation of the upgrade kit itself:

- Schema changes from the logical schema (in Siebel Tools) are automatically applied to the physical schema of the HQ database.
- The custom schema version number in the HQ database is automatically incremented by one.

This task is a step in *Process of Updating the Siebel Database Schema*.

The number and type of upgrade kits that you need for deploying Siebel database schema changes depend on the kinds of changes you made to the schema:

- **Major schema changes.** Siebel Anywhere is suitable for distributing minor Siebel database schema updates that have been created using database extensibility. If you are performing a major upgrade, then do not use Siebel Anywhere to distribute your changes.

  For more information about performing major upgrades, see *Siebel Database Upgrade Guide* . For more information about extracting database changes for Mobile Web Clients, see *Siebel Remote and Replication Manager Administration Guide* , and for detailed information regarding database extensibility, see *Configuring Siebel Business Applications* .
- **Incremental schema changes.** If your Siebel database schema changes are *incremental* (that is, if you have added tables, added columns, or changed the names of existing tables or columns), then you can create a Siebel Anywhere upgrade kit to upgrade Mobile Web Clients or a Regional Node Server. You do not need to reextract.
- **Preference for reextraction rather than Siebel Anywhere.** It is possible to reextract databases for Mobile Web Clients and Regional Databases, rather than updating them using Siebel Anywhere, if you prefer. For more information, see *Siebel Remote and Replication Manager Administration Guide* .
- **Mandatory reextraction for nonincremental schema changes.** If your Siebel database schema changes are *nonincremental* (that is, if you have removed or deactivated tables or columns, which are changes that are *not* recommended), then you *must* reextract all Mobile Web Clients and Regional Node Servers. Otherwise, synchronization might fail due to transactions that contain data for columns or tables that are no longer available. Reextraction makes it unnecessary to create a Siebel Database Schema upgrade kit.
- **Major schema changes.** Siebel Anywhere is suitable for distributing minor Siebel database schema updates that have been created using database extensibility. If you are performing a major upgrade, then do not use Siebel Anywhere to distribute your changes.
- For more information about performing major upgrades, see *Siebel Database Upgrade Guide* . For more information about extracting database changes for Mobile Web Clients, see *Siebel Remote and Replication Manager Administration Guide* , and for detailed information regarding database extensibility, see *Configuring Siebel Business Applications* .
- **Incremental schema changes.** If your Siebel database schema changes are *incremental* (that is, if you have added tables, added columns, or changed the names of existing tables or columns), then you can create a Siebel Anywhere upgrade kit to upgrade Mobile Web Clients or a Regional Node Server. You do not need to reextract.
- **Preference for reextraction rather than Siebel Anywhere.** It is possible to reextract databases for Mobile Web Clients and Regional Databases, rather than updating them using Siebel Anywhere, if you prefer. For more information, see *Siebel Remote and Replication Manager Administration Guide* .
- **Mandatory reextraction for nonincremental schema changes.** If your Siebel database schema changes are *nonincremental* (that is, if you have removed or deactivated tables or columns, which are changes that are *not* recommended), then you *must* reextract all Mobile Web Clients and Regional Node Servers. Otherwise, synchronization might fail due to transactions that contain data for columns or tables that are no longer available. Reextraction makes it unnecessary to create a Siebel Database Schema upgrade kit.

After you define the upgrade kit or upgrade kits for your upgrade, you must activate and apply them as described in *Activating, Applying, and Distributing Upgrade Kits* If you created both a Siebel Database Schema upgrade kit and a Siebel Repository Export Import upgrade kit, then make sure that both are activated and applied before you proceed to the next part of the process.

CAUTION: If you make any upgrade kits dependent upon the Siebel Database Schema component, then first-time Mobile Web Client users will receive an error when they initialize their local databases. The error will indicate the Siebel Business Applications was unable to check initialization status. This problem does not occur where the local database already exists.

Preparing Mobile Web Clients for Siebel Database Schema Changes

In either a test environment or a production environment, it is recommended that you have your Mobile Web Client users synchronize before you distribute your Siebel Database Schema upgrade kit.

This task is a step in *Process of Updating the Siebel Database Schema*.

Note: If any of your mobile users connect to a Regional Node Server, rather than the HQ Server, then the Regional Node Server must install the Siebel database schema changes before those mobile users can obtain and install the changes.

After Mobile Web Client users have synchronized, you can distribute your active upgrade kits to an appropriate test or production configuration for your Mobile Web Clients. For more information about distributing upgrade kits, see *Distributing Upgrade Kits*.

Preparing Siebel Regional Node Servers for Siebel Database Schema Changes

This topic describes how to prepare Siebel Regional Node Servers for Siebel database schema changes.

This task is a step in *Process of Updating the Siebel Database Schema*.

Perform this task before you distribute your Siebel Database Schema upgrade kit to either test Regional Node Servers or production Regional Node Servers.

To prepare Siebel Regional Node Servers for Siebel database schema changes

1. Have your Regional Node Servers synchronize with the HQ Server.
2. Determine which configuration you will use to distribute the upgrade kit or upgrade kits to your Regional Node Servers.
   The default configuration that ships with Siebel Business Applications is Siebel Regional Server. You can create your own configuration for Regional Node Servers to suit the particular needs of your organization. Make sure that the Siebel Database Schema component is associated with your custom configuration. For more information about these tasks, see *Identifying Configurations to Deliver Upgrade Components* and *Modifying and Creating Siebel Anywhere Configurations*.
3. Inspect the Upgrade Component server parameter value for each Regional Node Server, and change it if necessary. For more information, see *Checking Siebel Regional Node Server Parameters Before a Siebel Database Schema Update*.
4. Distribute your upgrade kit (or upgrade kits) to a configuration for your Regional Node Servers.

These might be either test Regional Node Servers, or production Regional Node Servers, depending on where you are in the process. For more information about distributing upgrade kits, see *Distributing Upgrade Kits*.

Checking Siebel Regional Node Server Parameters Before a Siebel Database Schema Update

This topic describes how to check Siebel Regional Node Server parameters before you distribute a Siebel Database Schema upgrade kit to a Regional Node Server.

This task is a step in *Process of Updating the Siebel Database Schema*.

Before you distribute a Siebel database schema to a Regional Node Server, you must make sure that the server parameter Upgrade Component on the Regional Node Server is set to the correct value to match either Siebel Regional Server or any custom configuration you have made for your Regional Node Server. The simplest way to modify the parameter is to use the Server Parameters view, as described in the following procedure.

CAUTION: Before installing a Siebel Database Schema upgrade kit on a Regional Node Server in an IBM DB2 environment, you must drop all customized views and triggers in the regional database. Otherwise, the Upgrade Wizard will encounter errors when applying the changes in the logical schema to the physical schema.

Setting Siebel Regional Node Server Parameters Before Distributing a Siebel Database Schema Update

The following procedure describes how to set Siebel Regional Node Server parameters before you distribute a Siebel database schema update.

To set Siebel Regional Node Server parameters before distributing a Siebel database schema update

1. On the Regional Node Server, Navigate to the Administration - Server Configuration screen, then the Servers view.

   The Siebel Servers list appears.
2. From the Siebel Servers list, select the appropriate server, and click the Parameters tab.

   The Server Parameters list and the Additional Parameter Information form appear.
3. In the Server Parameters list, select Upgrade Component in the Parameter field, and set Value to Deploying Siebel Open UI or the name of your custom configuration.

   The Value on Restart setting is automatically populated with the same value when you leave the Value field.
4. Still in the Server Parameters list, select Table Owner Password in the Parameter field, and make sure that Value and Value on Restart are not blank.

5. Still in the Server Parameters list, select Version Check in the Parameter field, and set Value and Value on Restart to True.

   Note: You can also check and modify the Upgrade Component, Table Owner Password, and Version Check parameters in Server Manager.

You can also check and modify Regional Node Server parameters using Server Manager, as described in the following two procedures.

Checking Server Parameters Using the Server Manager Command Line

The following procedure describes how to check server parameters using the Siebel Server Manager command line.

To check server parameters using the Server Manager command line

- From the command-prompt, enter:

  ```
  list param <UpgComponent, VersionCheck, TableOwnPass>
  ```

Modifying Server Parameters Using the Server Manager Command Line

The following procedure describes how to modify server parameters using the Siebel Server Manager command line.

To modify server parameters using the Server Manager command line

- Enter:

  `change param` param_alias = parameter_value

  where *param_alias* is UpgComponent, VersionCheck, or TableOwnPass.

For more information about other steps in the process of preparing a Regional Node Server for a Siebel database schema update, see *Preparing Siebel Regional Node Servers for Siebel Database Schema Changes*.

Installing Siebel Database Schema Changes on Mobile Web Clients

In a test environment, install the Siebel Database Schema upgrade kit (and any accompanying upgrade kits) on test Mobile Web Clients. In a production environment, install the Siebel Database Schema upgrade kit (and any accompanying upgrade kits) on production Mobile Web Clients. For more information, see *Retrieving, Installing, and Testing Upgrade Kits*

This task is a step in *Process of Updating the Siebel Database Schema*.

Note: The Siebel Database Schema upgrade kit will only be downloaded by Mobile Web Client users.

Installing the Siebel Database Schema upgrade kit synchronizes the logical and physical schemas on Mobile Web Client databases.

To distribute an upgrade kit for Mobile Web Clients

- For Mobile Web Clients assigned to the HQ Server, distribute the Siebel Database Schema upgrade kit under the appropriate configurations so each Mobile Web Client can receive the upgrade kit during synchronization.
- For Mobile Web Clients assigned to a Regional Node Server, distribute the Siebel Database Schema upgrade kit to the configuration for the Regional Node Server, and install the update on the Regional Node Server before distributing the upgrade kit to the configuration for that server's Mobile Web Clients.

  Siebel database schema changes must also be complete on a Regional Node Server before any Developer Web Clients connected to that server can resume normal operations.

Installing Siebel Database Schema Changes on Siebel Regional Node Servers

In a test environment, install the Siebel Database Schema upgrade kit (and any accompanying upgrade kits) on test Siebel Regional Node Servers. In a production environment, install the Siebel Database Schema upgrade kit (and any accompanying upgrade kits) on production Regional Node Servers. The following procedure describes how to install a Siebel database schema update on a Regional Node Server.

This task is a step in *Process of Updating the Siebel Database Schema*.

Installing the Siebel Database Schema upgrade kit synchronizes the logical and physical schemas on Regional Node Server databases.

Note: If any of your Mobile Web Client users connect to a Regional Node Server, rather than the HQ Server, then the Regional Node Server must install the Siebel database schema changes before those mobile users can obtain and install the changes.

To install a Siebel Database Schema upgrade kit for regional databases

1. Disconnect all users from the Regional Node Servers and the HQ Server.
2. Stop all tasks on the Siebel Servers except those for the Replication Agent, Transaction Processor, Router, and Transaction Merger components.
3. Wait until the replication backlog has been processed.

   Check the download and upload statistics in the Administration - Siebel Remote screen for each Regional Node. The three Last File and Last Transaction numbers match in the lower applet. Make sure that there are no outstanding transactions by checking that the `docking\inbox` and `docking\outbox` directories are empty on the regional Siebel Servers.

4. Verify that the server parameter Version Check is set to TRUE on the regional Siebel Server by using one of the following methods:
   - In the Server Parameters view, verify that the server parameter Version Check is set to TRUE on the regional Siebel Server.
   - Use the following commands to start srvrmgr and verify the value of the Version Check parameter:

     ```
     srvrmgr /g gateway_host /e enterprise_server /s server_name /u user_name /p password

     list param VersionCheck
     ```
5. The Replication Agent now performs a version check and shuts down itself and the regional Siebel Server so the Siebel administrator can perform the upgrade manually.
6. Check the appropriate log files (RepAgent.log and syncthrd.log) to verify whether you need to run the Upgrade Wizard.
7. Verify that the Siebel Server is properly shut down. If not, then shut it down manually.
8. Use one of the following methods to run the Upgrade Wizard:
   - In a Windows environment, in the server `BIN` directory, run siebupg.exe to perform the upgrade.
   - In a UNIX environment, in the server `BIN` directory, run srvrupgwiz to perform the upgrade.
9. After the upgrade, bring up the regional Siebel Server.
10. You can change the server parameter Version Check back to False, which is its default value.

> **Note:** When you install the Siebel Database Schema upgrade kit, the schema changes are applied to the server and the custom version is incremented, which makes the database template obsolete. If there are Mobile Web Client users who connect to the Regional Node Server, then you must run Generate New Database task to create a new database template.

Mobile Web Clients can synchronize and install the Siebel Database Schema upgrade kit in much the same manner as any other upgrade kits by using the Upgrade Wizard.

If the Siebel Business Applications developer created or modified applets or views to display data in new extension columns or new extension tables, then it is necessary to create and distribute the accompanying Siebel Repository Export Import upgrade kit. Installing the combination of the Siebel Database Schema upgrade kit and the Siebel Repository Export Import upgrade kit allows users to view all data after the schema upgrade.

After the Siebel database schema upgrade is finished, distribute the new Siebel Repository Export Import upgrade kit to its corresponding Mobile Web Clients.

Deleting Old Upgrade Kits

Sometimes you might want to purge older upgrade kits to free up some space used by these upgrade kits. When you purge old upgrade kits, you must make sure that users will not encounter errors when they retrieve upgrade kits that are expecting to use information from upgrade kits that have been deleted.

Two groups of users can be adversely affected when old upgrade kits are deleted, new users and infrequent users. For example, assume that your end users belong to the same configuration. Also assume they are currently using version 3 of Siebel Client Customer Revisions and version 4 of Siebel Repository Export Import.

If you were to delete all the previous Siebel Client Customer Revisions and Siebel Repository Export Import upgrade kits, then a new user might have trouble with Siebel Client Customer Revisions if you start from version 0. This situation occurs because Siebel Client Customer Revisions upgrade kits often depend on the previous versions. If a new user

needs version 2 in order to gain access to version 3, and if version 2 is deleted, then Siebel Anywhere would not be able to provide the new user with version 3. However, the new user *would* be able to install version 4 of the Siebel Repository Export Import, because Siebel Repository Export Import upgrade kits do not depend on any version history.

The correct way to handle this situation is to keep an updated installation for new users where each of the upgrade kits has been installed. You can do this by creating a package periodically and making the most current package available to new users. This method addresses the problem for new users, because they always install a client with up-to-date configurations.

However, do *not* delete old upgrade kits as soon as a new package is created. There might be infrequent users in your organization who have not logged in for months. While the rest of the company is using version 3, the infrequent users might still be on version 1. In this case, it is up to the individual Siebel administrators to assess the situation and determine how old the upgrade kit must be before deletion.

Sometimes you have a few upgrade kits of the type Customer Revision that were distributed long ago. Use the following procedure to delete them. Use the same logic if you are planning to delete any other upgrade kits.

To delete old Customer Revision upgrade kits

1. Establish one installation that includes all the Customer Revision upgrade kits ever installed.
2. Create a package from that installation.
3. Set up each new user with the package prepared in the previous Steps. This provides new clients with the files that were distributed through the old Customer Revision upgrade kits, and also provides new clients with the correct version of the clntrev.cfg file.

Testing the Consistency of Upgrade Kit Components

After you upgrade a Siebel application to a new release, check to make sure that version numbering continues for the releases for upgrade components in each Siebel Anywhere configuration. Upgrading to a new release of Siebel Business Applications upgrades the Siebel Anywhere table definitions. However, it does not reset the Upgrade Component information for Siebel Anywhere configurations.

As a result, Siebel Anywhere will resume version checking against the existing component versions in the new release of Siebel Business Applications. If required upgrade kits for Siebel Client Repository Export Import, Siebel Client Customer Revisions, or Siebel CFG File were distributed in the previous release, then newly upgraded production clients might be incorrectly prompted for upgrade kits when they synchronize with the server.

Before upgrading production users, test the upgrade kit on a Siebel administrator's client with version checking enabled. Verify that this client passes version check, so that after the upgrade kit is packaged and distributed, it will function correctly for the remainder of your production users.

To accomplish this testing, verify that the administrator's client has the latest versions for Siebel Repository Export Import, Siebel Client Customer Revisions, and Siebel CFG__*language-code*.

The following procedures describe how to test Siebel Client Customer Revisions and CFG file components for consistency.

To test Siebel Client Customer Revisions components for consistency

- Copy the clntrev.cfg file from one of your previous release production clients to the SIEBEL_CLIENT_ROOT `\BIN` directory of the Siebel administrator's client.

To test Siebel CFG file components for consistency

- Edit the appropriate CFG file under SIEBEL_CLIENT_ROOT `\BIN\` language-code and increment the Version parameter in the [Siebel] section to the Maximum Version of the Upgrade Component Siebel Call Center CFG file for the previous release.

Creating Upgrade Kits That Check DLL File Versions

Sometimes you might need to upgrade third-party software based upon the version of a particular DLL file. An example would be an ODBC upgrade on a client, based upon the version of the file odbcint.dll.

The following procedure describes how to create a custom component and an upgrade kit that check the version of a particular DLL file.

To create an upgrade component and upgrade kit for DLL version check

1. Navigate to the Administration - Siebel Anywhere screen, then the Upgrade Component List view.

   The Upgrade Components list appears
2. Create an upgrade component and specify the field values shown in the following table.

Field Name	Value
Name	*upgrade_component_name*
Component Type	Third Party Software
Locate Method	Registry
Locate Information	*hkey,subkey_name,value_name*
Version method	FileVersion

   In this table, the value for Locate Information points to the registry hkey, subkey and a specific value that points to the DLL file that you want to version check. For example:

   ```
   HKEY_LOCAL_MACHINE,SOFTWARE\ODBCFileDSN\DefaultIcon,Driver
   ```

   If the registry value is DefaultIcon, then the *value_name* must be omitted. This value in the registry will then point to the specific DLL file. For example:

   ```
   C:\WINNT\System32\odbcint.dll
   ```
3. After creating the custom component, as described in the previous steps of this procedure, define the upgrade kit as described in *Defining Upgrade Kits*

When defining the upgrade kit, remember to select the name of the custom upgrade component (created in the previous step) when the Upgrade Kit Wizard page appears.

Example of Constructing a Third Party Software Upgrade Kit

This topic provides an example of how to construct a Third Party Software upgrade kit, using WinZip as the example software for which the upgrade kit is created. The process shown in this topic can be used to construct upgrade kits for other third-party software applications, if the software is related to the use of Siebel Business Applications.

There are two procedures to this process:

- *Creating a Third Party - Win Zip Upgrade Component*
- *Defining a Third Party - Win Zip Upgrade Kit*

Creating a Third Party - WinZip Upgrade Component

The following procedure describes how to create a Third Party - WinZip upgrade component.

To create a new upgrade component Third Party - WinZip

1. Navigate to the Administration - Siebel Anywhere screen, then the Upgrade Component List view.

   The Upgrade Components list appears.
2. On the Upgrade Components list, click New.

   A blank record appears.

   Enter the necessary information.

   As an alternative, copy the record of an existing Third Party component and then edit the fields as required.
3. Enter the values shown in the following table.

Field Name	Value
Component Name	Third Party - WinZip
Component Type	Third Party Software
Locate Method	Registry
Locate Information	HKEY_CLASSES_ROOT,WinZip\shell\open\command

Field Name	Value
	Locate Information has the following structure when Locate Method is set to Registry: *HKEY*, *Subkey*,*Name*. If *Name* is equal to the default, that is, (Default) as seen in the Registry Editor, then leave *Name* as blank for Locate Information.
Version method	FileVersion

4. From the link bar, select Upgrade Configurations.

   The Upgrade Configurations list appears, along with the Upgrade Components list.
5. On the Upgrade Configurations list, select the desired clients such as Siebel Sales Client.
6. On the Upgrade Components list, click New.

   The Upgrade Components pick list appears.
7. On the Upgrade Components Selection dialog box, select Third Party - WinZip.

   The Upgrade Configurations list appears, along with the Upgrade Components list.

   The new record (Third Party - WinZip upgrade component) is highlighted in the Upgrade Components list, which indicates that it is selected.

   This step completes the procedure to create a new upgrade component, Third Party - WinZip.

Defining a Third Party - WinZip Upgrade Kit

The following procedure describes how to define a Third Party - WinZip upgrade kit.

To define a Third Party - WinZip upgrade kit

1. Navigate to the Administration - Siebel Anywhere screen, then the Upgrade Kits view.

   The Upgrade Kits view appears.
2. Click Auto Create to invoke the Upgrade Kit Wizard.

   The Upgrade Kit Wizard - Upgrade Components page appears.
3. Choose Third Party - WinZip in the list and click Next.

   The Upgrade Kit Wizard - Add Files page appears.
4. Input the UNC path of the WinZip installer.
5. Click Next.

   The Upgrade Kit Wizard - Select Execute File page appears.
6. Input the executable file name and the command-line argument, if any.
7. Follow the prompts to perform the following actions:
   a. Review and verify the information.
   b. Specify the new version number.
   c. (Optional) Change the title, if you want to do so.

8. When you have finished entering upgrade kit information, click Finish.

   Follow the procedures in *Activating, Applying, and Distributing Upgrade Kits* to activate, apply, and distribute the upgrade kit.

8 Troubleshooting for Siebel Anywhere

Troubleshooting for Siebel Anywhere

This chapter includes tips to help Siebel administrators resolve some problems that can arise while working with Siebel Anywhere. It includes the following topics:

- *Upgrade Kit Wizard Error at Finish*
- *New Upgrade Kit Status: Request Submitted*
- *New Upgrade Kit Status: In Progress*
- *Unable to Download Upgrade Kit*
- *Required Upgrade Kit Cannot Be Installed*
- *Failed Upgrade Kit Locks the Siebel Administrator's Account*
- *Converting a Required Upgrade Kit to an Optional Upgrade Kit*
- *Changing Version Settings After Distribution*

Note: For any troubleshooting that involves Mobile Web Clients, make sure that the Transaction Processor and Transaction Router components are running on the Siebel Server with which the mobile users synchronize. Verify the status of the process. For more information, see *Siebel Remote and Replication Manager Administration Guide* .

Upgrade Kit Wizard Error at Finish

If you receive an error message such as `Unable to find definition for component UpgKitBldr` when you click Finish in the Upgrade Kit Wizard, then you might need to synchronize server components. For more information, see *Synchronizing Server Components*.

New Upgrade Kit Status: Request Submitted

Occasionally, you might notice that the status of a new upgrade kit remains Request Submitted after several minutes of waiting. This status means that the Upgrade Kit Builder has not yet picked up the server request to build the upgrade kit. There are several possible causes for this delay:

- The upgrade kit might have been created when the Siebel Anywhere component was not enabled. The status will not change until the component is enabled. After the component is enabled, the request will be processed automatically.
- The server component Upgrade Kit Builder (alias UpgKitBldr) might be busy building a previously submitted upgrade kit. Most upgrade kits take only a few minutes to build. However, Siebel Database Schema upgrade

kits can take as long as two hours or more to build, depending on the database type, the database size, and the extent of the schema changes.

- The server component Upgrade Kit Builder (alias UpgKitBldr) might not be running.
- The server component Server Request Broker (alias SRBroker) might not be running correctly.
- The server component Server Request Processor (alias SRProc) might not be running correctly.

The following paragraphs provide information about troubleshooting some of these problems.

Checking for Current Upgrade Kit Builder Activity

The following procedures provide ways to check whether the Upgrade Kit Builder is busy building another upgrade kit:

To check Upgrade Kit Builder activity in a Siebel application

1. Navigate to the Administration - Server Management screen, then the Components view.
2. In the Components list, select the Upgrade Kit Builder component.
3. Click the Tasks view tab, and inspect the status of any tasks submitted for the Upgrade Kit Builder component.

To check Upgrade Kit Builder activity using srvrmgr

- In srvrmgr, enter the following command to check the status of Upgrade Kit Builder tasks:

```
srvrmgr> list task for comp UpgKitBldr
```

Checking the Status of Components Related to Upgrade Kits

Use any of the following ways to check the status of the UpgKitBldr, SRBroker, and SRProc components:

- Navigate to the Administration - Server Management screen, then the Components view. Select each applicable component, and click the Tasks view tab to inspect the status of tasks submitted for that component.
- In srvrmgr, use the following command to view the status of a component, replacing *component_name* with UpgKitBldr, SRBroker, or SRProc:

  ```
  srvrmgr> list comp component_name
  ```

- In the log directory, check the log files as follows:
  - Inspect the *enterprise_server_name.siebel_server_name*.log to verify that all the processes for UpgKitBldr, SRBroker, and SRProc were created correctly.
  - See whether there are any signs of trouble in the SRProc_*task#*.log file or the SRBroker_*task#*.log file.

Troubleshooting Server Request Processor Problems

If you find evidence that the SRProc component is not running correctly, then check the following aspects of your server's configuration:

- Is the database connection information correct?

- Are the user name and password correct?
- Have you done server component synchronization? For more information about this procedure, see *Siebel System Administration Guide* .

For more information about troubleshooting server problems, see *Siebel System Monitoring and Diagnostics Guide* .

New Upgrade Kit Status: In Progress

Occasionally, you might notice that the status of a new upgrade kit remains In Progress for an extended period, indicating that the upgrade kit is taking a long time to build. If this is the case, then consider the following information to help troubleshoot the situation.

Some of the reasons for a delay in status change include:

- **The upgrade kit is still being built.** (In particular, Siebel Database Schema upgrade kits can take as long as two hours or more to build.) To monitor the progress of the upgrade kit, use one of the following options to check the latest status:
  - Navigate to the Administration - Server Management screen, then Components, and then Tasks. Look for the latest status of the Upgrade Kit Builder component.
  - In the server manager line-mode utility, enter:

    ```
    list task for comp UpgKitBldr
    ```
  - Look for the component log file for Upgrade Kit Builder (UpgKitBldr_taskno.log) in the log directory.

- **Upgrade Kit Builder is not responding.** Sometimes a Siebel File System error or a network connection error interrupts the operation of Upgrade Kit Builder. If you conclude that Upgrade Kit Builder is not responding, then take the following actions:
  - Check these server components, which support Upgrade Kit Builder: File System Manager (FSMSrvr) and Server Request Broker (SRBroker). Make sure that these components are running correctly.
  - Edit the record, change the status from In Progress to Error, save the record, and then delete the record before trying to rebuild. (The reason for deleting the failed record is to avoid a naming conflict.) Then, recreate the upgrade kit.

Unable to Download Upgrade Kit

The following error messages sometimes appear when you attempt to download an upgrade kit:

- Unable to find upgrade kit path to upgrade component: *component_name*. Please contact your Siebel administrator or try again.
- The file "*upgrade_kit_file_name.arc*" could not be found on any specified file system.
- Upgrade wizard did not complete successfully. Restart Siebel application to attempt the upgrade again.
- Unable to open compressed file
- (*Sublevel_directory_path*\S_UPG_KIT_IARG_*ROWID_REVNO*.saf).
- Unable to download the Upgrade Kit "*upgrade_kit_title*" required to upgrade your system.

These errors can be caused by the following conditions:

- **A previously required upgrade kit has been deactivated or deleted.** The client is trying to find an active upgrade kit with a certain New Version number, but there is no such upgrade kit because it was deactivated or deleted.

  To solve this problem, have your Mobile Web Client first synchronize using an invalid Configuration (set the Component Name in the CFG file to none). This action will get the recent transactions routed to the Mobile Web Client and pass the transaction regarding the nonfunctioning or deactivated upgrade kit.

  Then have users change the Component Name back to the original value and synchronize again. You can make this process transparent to your users by sending them a batch file that synchronizes for them from the command line, using a different CFG file that was sent to them with the batch file. Use siebsync.exe and specify the CFG file with the /c option.
- **A nonexecutable file has been specified for execution.** The upgrade kit you have created is of type Siebel Client Customer Revisions and when you were creating the upgrade kit, you selected a nonexecutable file to be executed. You must deactivate the upgrade kit and recreate it with the same version number. Activate the upgrade kit, but do not apply or distribute it.
- **A user's connection to the Siebel File System is not set correctly.** Use one of the following techniques, depending on the type of user involved:
  - **For Developer Web Client users.** Check the Developer Web Client users' CFG files and make sure that the FileSystem parameter under the [ServerDataSrc] section is set to the appropriate location and that your user has read and write access to that directory. One test for correct location and read and write access would be to have the user open an attachment. If the attachment opens, then the connection to the Siebel File System is set correctly.
  - **For Mobile Web Client users.** Because Mobile Web Client users are connected to the Siebel File System through the Siebel Server, make sure that the Siebel File System parameter of Synchronization Manager is set to the correct location: the same location as the FileSystem parameter in the CFG file (of the user who has created the upgrade kit). In other words, make sure that the File System Manager is working properly.
- **There is not enough space on the computer to download the upgrade kit.** Free some space and try to retrieve and install the upgrade kit again.
- **A prerequisite upgrade kit is not available.** An upgrade that the Siebel administrator intends for the end users to retrieve might depend on another upgrade kit that has either been deactivated, deleted, or not created yet. The Siebel client will make sure that all upgrade kits are ready before invoking the Upgrade Wizard.

Required Upgrade Kit Cannot Be Installed

If you create and distribute a required upgrade kit that, for some reason, cannot be installed on the client or server, then you must define and distribute a repair upgrade kit, which is a new upgrade kit that functions properly.

Use the Upgrade Kit Wizard to define the repair upgrade kit. Make sure that the version information in the repair upgrade kit differs from the version information in the nonfunctioning upgrade kit in the following ways:

- If the minimum version is editable, then make sure that its value is set to the last known good version for this upgrade component.

  For example, if the minimum version of the nonfunctioning upgrade kit was 5, then set the minimum version of the repair upgrade kit no higher than 5.
- Make sure that the new version of the repair upgrade kit is at least one version number higher than the new version of the nonfunctioning upgrade kit.

  For example, if the new version of the nonfunctioning upgrade kit was 6, then make the new version of the repair upgrade kit 7. Then, Oracle's Siebel Anywhere can upgrade the user's setup from the last good version, skipping the nonfunctioning version.

After you have defined your repair upgrade kit, activate and test the upgrade kit, as you would do for any other required upgrade kit. Distribute the repair upgrade kit to the appropriate Developer or Mobile Web Client users, as described in the following procedure.

To distribute a repair upgrade kit

1. Navigate to the Administration - Siebel Anywhere screen, then the Upgrade Configurations view.

   The Upgrade Configurations list appears.
2. Prepare the upgrade kit for distribution.
   a. In the Upgrade Configurations list, select the configuration under which you distributed the nonfunctioning upgrade kit.
   b. Verify that the components in the repair upgrade kit are included in the configuration as related components.
   c. With this configuration selected, click Distribute.

Failed Upgrade Kit Locks the Siebel Administrator's Account

If the Siebel administrator's account is locked out due to a failed upgrade kit, then the following procedure will help you recover.

It is also sometimes possible to log in as a Web Client that does not have version checking or upgrades, and take corrective measures to resolve the issue related to the failed upgrade.

To unlock the Siebel administrator's account

1. Edit the Siebel administrator's CFG file by changing the value of the ComponentName parameter to None.

   This change prevents the Siebel client from performing any version checking. If you dynamically associated the Siebel administrator account with a configuration, then create a service request (SR) on My Oracle Support.

Alternatively, you can phone Oracle Global Customer Support directly to create a service request or get a status update on your current SR. Support phone numbers are listed on My Oracle Support.

> **CAUTION:** Dynamically associating the Siebel administrator account with any configuration is not recommended, because it sometimes prevents the administrator from logging in.

2. If the upgrade kit failed while the Upgrade Wizard was running, then delete the file upgwiz.ucf in the `BIN` directory.

   The Siebel client always checks for the existence of the file upgwiz.ucf in the `BIN` directory to find out if it was in the middle of an upgrade.
3. Log in to the Siebel Business Applications to determine the reason for the failed upgrade, and take corrective measures.

   Presumably, the other subscribers to your original configuration are also affected.
4. Test your solutions using the Siebel Test Client configuration.
5. After testing the upgrade kit thoroughly, distribute the correct upgrade kit to every appropriate configuration.

   Remember to change the value of the ComponentName parameter to its original value in the CFG file.

Converting a Required Upgrade Kit to an Optional Upgrade Kit

If you distribute a required upgrade kit for Developer Web Client users and realize it was supposed to have been optional, then use the following procedure to convert the upgrade kit from required to optional.

> **Note:** *This procedure applies only to Developer Web Client users.* Upgrade kits for Mobile Web Client users cannot be changed from required to optional. If a Mobile Web Client user has difficulty with a required upgrade kit, then you must deactivate the problematic upgrade kit and create a new one. The new upgrade kit will have the same Minimum Old Version and Maximum Old Version settings as the deactivated upgrade kit, but a higher New Version number.

To convert a required upgrade kit to an optional upgrade kit (for Developer Web Clients only)

1. Navigate to the Administration - Siebel Anywhere screen, then the Upgrade Component List view.

   The Upgrade Components list appears.
2. From the Upgrade Components list, select the name of the Upgrade Component for which you distributed the upgrade kit.
3. Modify the Min Version field with the minimum value that is acceptable, or change it to null.

   > **CAUTION:** Perform this step carefully, because the wrong minimum value could cause some users to skip a related required upgrade kit.

4. Step off the record, or choose Save Record from the applet menu, to save the record.
5. Navigate to the Administration - Siebel Anywhere screen.
6. From the link bar, select Upgrade Configurations.

   The Upgrade Configurations and Upgrade Components list applets appear.

7. On the Upgrade Configurations list applet, select the configurations you had previously distributed the component information to.
8. Click Distribute.

This procedure makes the upgrade kit optional. Siebel Developer Client users and Mobile Web Client users can log in to the Component Upgrades view if the client version they are using is between the minimum and maximum for the upgrade component in the Upgrade Component List view.

Changing Version Settings After Distribution

If users receive an error message such as `Unable to select required upgrade kits to upgrade your system` when attempting to retrieve an upgrade kit, then Minimum Old Version and Maximum Old Version might be set inappropriately.

The following procedure describes how to correct inappropriate values for Minimum Old Version and Maximum Old Version after an upgrade kit has been distributed.

To change Minimum Old and Maximum Old versions of an upgrade kit after distribution

1. Navigate to the Administration - Siebel Anywhere screen, then the Upgrade Kits view.

   The Upgrade Kits list appears.
2. From the Upgrade Kits list, select the target upgrade kit.
3. If the status is Active, then change the status to Pending.
4. Change Minimum Old Version and Maximum Old Version values according to your requirement.

   If you want the upgrade kit to be available for every client, then set both values to blank.
5. While the upgrade kit is selected, click Activate.

   > **Note:** Do not change the status of the upgrade kit to Active manually. Manually changing the status in this way does not actually activate the upgrade kit.

You do not need to reapply or redistribute the upgrade kit.