



UPGRADE GUIDE FOR MICROSOFT WINDOWS

MIDMARKET EDITION

VERSION 7.5

12-BCID65

SEPTEMBER 2002

Siebel Systems, Inc., 2207 Bridgepointe Parkway, San Mateo, CA 94404
Copyright © 2002 Siebel Systems, Inc.
All rights reserved.
Printed in the United States of America

No part of this publication may be stored in a retrieval system, transmitted, or reproduced in any way, including but not limited to photocopy, photographic, magnetic, or other record, without the prior agreement and written permission of Siebel Systems, Inc.

The full text search capabilities of Siebel eBusiness Applications include technology used under license from Hummingbird, Ltd. and are the copyright of Hummingbird Ltd. and/or its licensors.

Siebel, the Siebel logo, TrickleSync, TSQ, Universal Agent, and other Siebel product names referenced herein are trademarks of Siebel Systems, Inc., and may be registered in certain jurisdictions.

Other product names, designations, logos, and symbols may be trademarks or registered trademarks of their respective owners.

U.S. GOVERNMENT RESTRICTED RIGHTS. Programs, Ancillary Programs and Documentation, delivered subject to the Department of Defense Federal Acquisition Regulation Supplement, are “commercial computer software” as set forth in DFARS 227.7202, Commercial Computer Software and Commercial Computer Software Documentation, and as such, any use, duplication and disclosure of the Programs, Ancillary Programs and Documentation shall be subject to the restrictions contained in the applicable Siebel license agreement. All other use, duplication and disclosure of the Programs, Ancillary Programs and Documentation by the U.S. Government shall be subject to the applicable Siebel license agreement and the restrictions contained in subsection (c) of FAR 52.227-19, Commercial Computer Software - Restricted Rights (June 1987), or FAR 52.227-14, Rights in Data—General, including Alternate III (June 1987), as applicable. Contractor/licensor is Siebel Systems, Inc., 2207 Bridgepointe Parkway, San Mateo, CA 94404.

Proprietary Information

Siebel Systems, Inc. considers information included in this documentation and in Siebel eBusiness Applications Online Help to be Confidential Information. Your access to and use of this Confidential Information are subject to the terms and conditions of: (1) the applicable Siebel Systems software license agreement, which has been executed and with which you agree to comply; and (2) the proprietary and restricted rights notices included in this documentation.

Contents

Introduction

How This Guide Is Organized	14
Revision History	14

Chapter 1. Planning an Upgrade

Naming and Typographical Conventions in This Guide	17
Overview of the Upgrade Process	18
Supported Upgrade Paths	22
Nonstandard Upgrades and Migrations	22
Upgrades from Release 7.0.x to Release 7.5	23
Migrations to Unicode	25
Before You Start	25
Useful Resources	26

Chapter 2. Data Migration Considerations

Changes to the Siebel Data Model	28
Access Control	28
Categorization	29
Party Model	31
Reviewing Columns Added to the S_SRC_PAYMENT Table	34
Data Migration Considerations for Release 7.x Applications	35
Activities	35
Calendar	36
Siebel Call Center	37

Calendar/Activities	38
Siebel eChannel	38
Siebel eMail Response	39
Expense Reports—Hotel Lines	40
Workflows	41
Service	42
Siebel Campaigns	42
Orders	44
Siebel Professional Services Automation (Siebel PSA)	46
Siebel Sales	52
Time Sheet	55
Universal Time Coordinated (UTC)	55

Chapter 3. Development Environment Preupgrade Tasks

Before You Begin	58
Preparing the Development Environment Database for Upgrade	60
Verifying Database Sort Order	60
Saving All Pending Changes	63
Backing Up the Development Database	63
Saving Interface Table Data	64
Identifying and Resolving Duplicates	64
Preparing Mobile and Dedicated Users for the Upgrade	65
Upgrading Your RDBMS Software	66
Verifying Database Server Configuration	68
Microsoft SQL Server Database Server Configuration	70
Oracle Database Server Configuration	73
DB2 Universal Database Server Configuration	76
Preparing the Database for a DB2 Upgrade	85
Creating DB2 Temporary Table Spaces and Buffer Pools	85
Increasing DB2 4-KB Table Space	86
Creating DB2 16-KB Table Space and Buffer Pool	86

Increasing DB2 16-KB Table Space	87
Creating DB2 32-KB Table Space and Buffer Pool	87
Increasing DB2 32-KB Table Space	88
Determining Table Space Page Size Requirements for DB2 UDB	88
Verifying Installation of the DB2 UDB Application Development Client	91
Preparing Tables for Upgrade	92
Preserving Custom Indexes on Tables	92
Considerations for Clustered Indexes	92
Disabling Customized Triggers	93
Dropping Customized Views	93
Identifying DB2 UDB Long Columns for Truncation	93
Preparing the Siebel Repositories for Upgrade	95
Renaming Your Existing Development Repository	96
Recording Dock Objects and Visibility Rules	97
Increasing Database File Size	97
Preparing Application Data for Upgrade	98
Siebel eChannel	98
Quotes	98
Timesheet	99
Calendar	100

Chapter 4. Upgrading the Development Environment

Installing Siebel Tools on the Development Workstations	105
Upgrading Gateway Name Server and Siebel Servers	106
Upgrading the Servers on the Windows Platform	107
Installing and Configuring the Siebel Database Server Software	109
Installing the Siebel Database Server Software	109
Installing Additional Languages for a Multilingual Deployment	110
Configuring the Siebel Database Server for Upgrade	110

Upgrading the Siebel Database Schema	118
Reviewing the Upgrade Log Files	119
Troubleshooting the Siebel Database Schema Upgrade	121
Restarting the Upgrade	122
Taking a Backup of the Database Repository	123
Adding New License Keys	123
Preparing the Prior Customer Repository for the Merge	124
Migrate Strings, Merge Labels and Fields, and Merge Templates	125
Automatic Upgrade of Copied Objects	127
Configuration Steps for Upgrade Inheritance	131
Preparing Customized Objects for the Merge	137
Identifying Changed Objects	137
Renaming Changed Objects	140
Performing a Repository Merge, Using Siebel Tools	143
Conflict Resolution	152
Troubleshooting	155
Generating EIM Temporary Columns	155
Updating User Responsibilities	156
Upgrading the Custom Database Schema	161
Troubleshooting the Custom Database Schema Upgrade	168
Restarting the Upgrade	168
Migrating Custom Business Component Configurations	170

Chapter 5. Development Environment Postupgrade Tasks

Postupgrade Tasks for All RDBMS Development Environments	173
Dropping DB2 8-KB Table Spaces and Buffer Pools	173
Manually Archiving the Log Files	173
Updating the File System Directory	174
Updating File System Attachments	175
Reapplying Custom Extensions on Obsolete Tables	177
Resolving Business Component and Join Conflicts	180

Identifying Obsolete Objects	181
Setting Visibility Modes for Access Control	183
Resolving Duplicate EIM Mappings	186
Safeguarding the New Custom Repository Export File	187
Producing a New Custom Configuration File	188
Regenerating the Database Template File	189
Extracting Developers or Siebel Tools Clients	189
Testing the Upgraded Configuration	189
Validating the Upgrade	190
Run Statistics	191
Reorganize the P1 Index for DB2 UDB	191
Reset Database Server Configuration Parameters	191
Clean Up Your Repository Files	191
Additional Postupgrade Tasks for Specific Applications	192
Call Center	192
Enterprise Application Integration (EAI)	193
Quotes	193
Siebel Workflow	194
Upgrading to RC2 Encryption	195
Prerequisites	196
Modifying the Input File	196
Changing User Properties	197
Changing the Password or Adding a New Key to a Database	198
Upgrading Your Encryption Method	199
Postupgrade Tasks for Global Deployments	202
Setting Up Your Environment to Support Global Time Zone	202
Migrating to Unicode	205

Chapter 6. Migrating the Client Configuration to the Web Client

About Migrating to the Siebel Web Client	208
Migration Approach	210
Applets and Views	210
Scripts	211
Client-Side Interfaces	214
Migration Tasks	215
Exposing Hidden Properties	215
Migrating Applets and Views	216
Reviewing Applet Migration	228
Remigrating Applets	229
Reviewing Modified Applet Migration	230
Reviewing View Migration	231
Remigrating Views	231
Reviewing Modified View Migration	232
Migration of Scripts	233
Applets	235
Business Components, Business Services, and Application Scripts	238
Migration of Client-Side Interfaces	240

Chapter 7. Production Environment Preupgrade Tasks

Before You Begin	246
Preparing the Production Environment Database for Upgrade	248
Preparing the Custrep.dat File	248
Verifying Database Sort Order	248
Backing Up the Production Database	251
Saving Interface Table Data	251
Identifying and Resolving Duplicates	251
Preparing Mobile and Dedicated Users for the Upgrade	252
Upgrading Your RDBMS Software	253

Verifying Database Server Configuration	254
Microsoft SQL Server Database Server Configuration	256
Oracle Database Server Configuration	259
DB2 Universal Database Server Configuration	261
Preparing the Database for a DB2 Upgrade	271
Creating DB2 Temporary Table Spaces and Buffer Pools	271
Increasing DB2 4-KB Table Space	272
Creating DB2 16-KB Table Space and Buffer Pool	272
Increasing DB2 16-KB Table Space	273
Creating DB2 32-KB Table Space and Buffer Pool	273
Increasing DB2 32-KB Table Space	274
Determining Table Space Page Size Requirements for DB2 UDB	274
Verifying Installation of the DB2 UDB Application Development Client . . .	277
Preparing Tables for Upgrade	278
Preserving Custom Indexes on Tables	278
Considerations for Clustered Indexes	278
Disabling Customized Triggers	279
Dropping Customized Views	279
Identifying DB2 UDB Long Columns for Truncation	279
Preparing the Siebel Repositories for Upgrade	281
Renaming Your Existing Production Repository	281
Recording Dock Objects and Visibility Rules	282
Increasing Database File Size	282
Preparing Application Data for Upgrade	283
Siebel eChannel	283
Quotes	284
Timesheet	285
Calendar	285

Chapter 8. Upgrading the Production Environment

Upgrading Gateway Name Server and Siebel Servers	289
Upgrading the Servers on the Windows Platform	290
Installing the Siebel Database Server Software	292
Upgrading Without a Development Environment	294
Installing and Configuring the Siebel Database Server Software	296
Installing the Siebel Database Server Software	296
Configuring the Siebel Database Server for Upgrade	297
Upgrading the Siebel Database Schema	305
Reviewing the Upgrade Log Files	306
Troubleshooting the Siebel Database Schema Upgrade	306
Restarting the Upgrade	307
Upgrading the Custom Database Schema	309
Troubleshooting the Custom Database Schema Upgrade	316
Adding New License Keys	318

Chapter 9. Production Environment Postupgrade Tasks

Postupgrade Tasks for All RDBMS Production Environments	321
Dropping DB2 8-KB Table Spaces and Buffers	321
Manually Archiving the Log Files	321
Updating the File System Directory	322
Updating File System Attachments	322
Generating Reporting Relationships	323
Setting Visibility Modes for Access Control	325
Producing a New Custom Configuration File	325
Validating the Upgrade	326
Run Statistics	326
Reorganize the P1 Index for DB2 UDB	326
Reset Database Server Configuration Parameters	326

Additional Postupgrade Tasks for Specific Applications	327
Call Center	327
Siebel Purchase Orders	328
Quotes	329
Siebel Workflow	329
Upgrading to RC2 Encryption	330
Postupgrade Tasks for Global Deployments	331
Setting Up Your Environment to Support Global Time Zone	331
Migrating to Unicode	332
Upgrading Mobile and Connected Clients	332

Appendix A. Upgrade Planning Worksheet

Worksheet	334
---------------------	-----

Index

Introduction

This guide contains guidelines and specific instructions for upgrading your Siebel eBusiness Applications, MidMarket Edition, to the latest release, Siebel 7.

This book will be useful primarily to people whose titles or job descriptions match one of the following:

Siebel System Administrators	Persons responsible for administering the whole system, including installing, maintaining, and upgrading Siebel products.
Database Administrators	Persons who administer the database system, including data loading; system monitoring, backup, and recovery; space allocation and sizing; and user account management.

NOTE: All Siebel MidMarket product names include the phrase MidMarket Edition to distinguish this product from other Siebel eBusiness Applications. However, in the interest of brevity, after the first mention of a MidMarket product in this document, the product name will be given in abbreviated form. For example, after Siebel Call Center, MidMarket Edition, has been mentioned once, it will be referred to simply as Siebel Call Center. Reference to a product using an abbreviated form should be understood as a specific reference to the associated Siebel MidMarket Edition product, and not any other Siebel Systems offering. When contacting Siebel Systems for technical support, sales, or other issues, note the full name of the product to ensure its proper identification and handling.

How This Guide Is Organized

This guide provides required information for upgrading Siebel eBusiness Applications. Its organization provides a logical and sequential explanation of the steps necessary to upgrade Siebel software, as described below:

- Upgrading the Development Environment
 - Upgrading Siebel Database Schema
 - Repository Merge
 - Upgrading Custom Database Schema
- Migration of Client Configuration from Release 6.x Windows Client to Release 7.x Web Client
- Upgrading the Production Environment
 - Upgrading Siebel Database Schema
 - Upgrading Custom Database Schema

The chapters appear in the order in which the upgrade should be conducted. To make the best use of this guide, carefully read [Chapter 1, “Planning an Upgrade”](#) and [Chapter 2, “Data Migration Considerations,”](#) first to understand the changes that occur during the Release 7.5 upgrade. Progress through the chapters sequentially, since each chapter builds on the preceding ones.

NOTE: Your Siebel implementation may not have all the features described in this guide, depending on which software modules you have purchased.

Revision History

Upgrade Guide for Microsoft Windows, MidMarket Edition, Version 7.5

This chapter provides instructions for upgrading to Release 7.5 from several prior releases of Siebel eBusiness Applications on Siebel-supported database platforms. While the general upgrade procedure is the same for all releases and platforms, the exact steps may vary slightly depending on the database platform and release from which you are upgrading.

Where the upgrade process differs from a platform standpoint, the proper steps for each release and database platform are clearly identified within this guide. For information about supported upgrade paths, see *Siebel System Requirements and Supported Platforms* on Siebel SupportWeb at <http://ebusiness.siebel.com/supportweb/>.

A successful upgrade requires:

- Familiarity with the basic conventions of the Microsoft Windows operating system under which your Siebel Servers will run.
- Familiarity with your specific Siebel eBusiness Applications deployment.
- Expertise in network connectivity, disk and file sharing, and software installation on your chosen application server and client operating systems.
- User accounts with administration privileges on Microsoft Windows.
- Expertise in database installation, tuning, and administration in your chosen relational database management system (RDBMS).
- Familiarity with the Siebel Packager Utility. For more information about the Siebel Packager Utility, see *Siebel Anywhere Administration Guide, MidMarket Edition* and *Siebel Web Client Administration Guide, MidMarket Edition*.

- Detailed understanding of customizations to support previous implementation of Siebel eBusiness Applications.

NOTE: This document explains how to upgrade your Siebel eBusiness Applications on several different database platforms, operating system platforms, and application server platforms. However, certain database and operating system platforms may not have been supported in previous releases, or may have been discontinued in the current release. Certain combinations of database and operating system platforms may also not be supported.

For this reason, information may appear in this manual about one or more platforms or combinations of database platforms and operating systems that are not currently available. For a list of currently supported systems, see *Siebel System Requirements and Supported Platforms* on Siebel SupportWeb at <http://ebusiness.siebel.com/supportweb/>.

Naming and Typographical Conventions in This Guide

This guide follows several naming conventions:

- Release 7.5 refers to the current release of the Siebel eBusiness Applications.
- Release 7.0.x refers collectively to all versions of Siebel 7.0 that are supported in upgrade to Release 7.5; for example, Release 7.0.3 or Release 7.0.4. Refer to the Upgrade section in *Siebel System Requirements and Supported Platforms* on Siebel SupportWeb at <http://ebusiness.siebel.com/supportweb/> for specific information on which release numbers are meant by Release 7.0.x.
- Release 7.x refers collectively to all versions of Siebel 7; for example, Release 7.0.3, Release 7.0.4, or Release 7.5. Refer to the Upgrade section in *Siebel System Requirements and Supported Platforms* on Siebel SupportWeb at <http://ebusiness.siebel.com/supportweb/> for specific information on which release numbers are meant by Release 7.x.
- Release 6.x refers collectively to all versions of Siebel 6 that are supported in upgrade to Release 7.5.
- Release 5.x refers collectively to all versions of Siebel 5 that are supported in upgrade to Release 7.5.
- The term “Windows” refers to all Microsoft Windows operating systems listed as supported for this release in *Siebel System Requirements and Supported Platforms*. Likewise, “MS SQL Server” refers to the version of that database referenced in *Siebel System Requirements and Supported Platforms* on Siebel SupportWeb at <http://ebusiness.siebel.com/supportweb/>.
- `SIEBEL_ROOT` refers to the directory where you have installed the new Siebel Server software.
- `DBSRVR_PLTFRM_ROOT` refers to the platform-specific subdirectory of the new Release 7.5 Database Server software. Under the root installation directory of the Database Server software is a subdirectory for each database platform. This subdirectory contains the actual files that will be executed during the upgrade process. Be sure to use the correct files for your database platform.
- `SIEBEL_CLNT_ROOT` refers to the directory where the Siebel Mobile Web Client is installed.

Unless otherwise noted, all references to Siebel eBusiness Applications documentation refer to the manuals provided with your Release 7.5 software.

Overview of the Upgrade Process

The upgrade should be conducted in the sequence in which it appears in this book, as illustrated in [Figure 1](#).

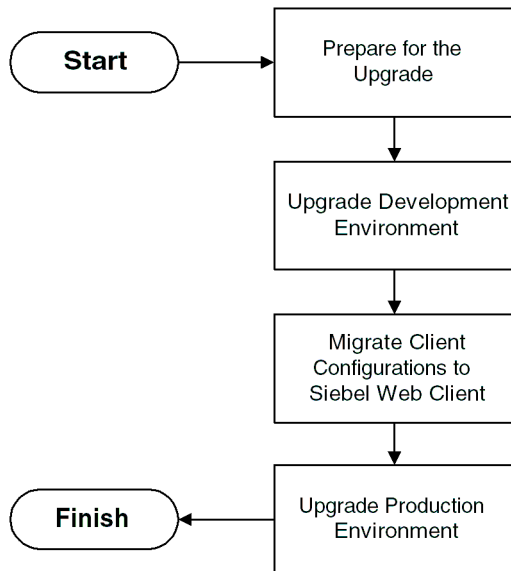


Figure 1. The Upgrade Process

[Figure 2 on page 19](#) illustrates the high-level steps that you need to perform for a development environment upgrade.

[Figure 3 on page 20](#) illustrates the high-level steps that you need to perform for a production environment upgrade.

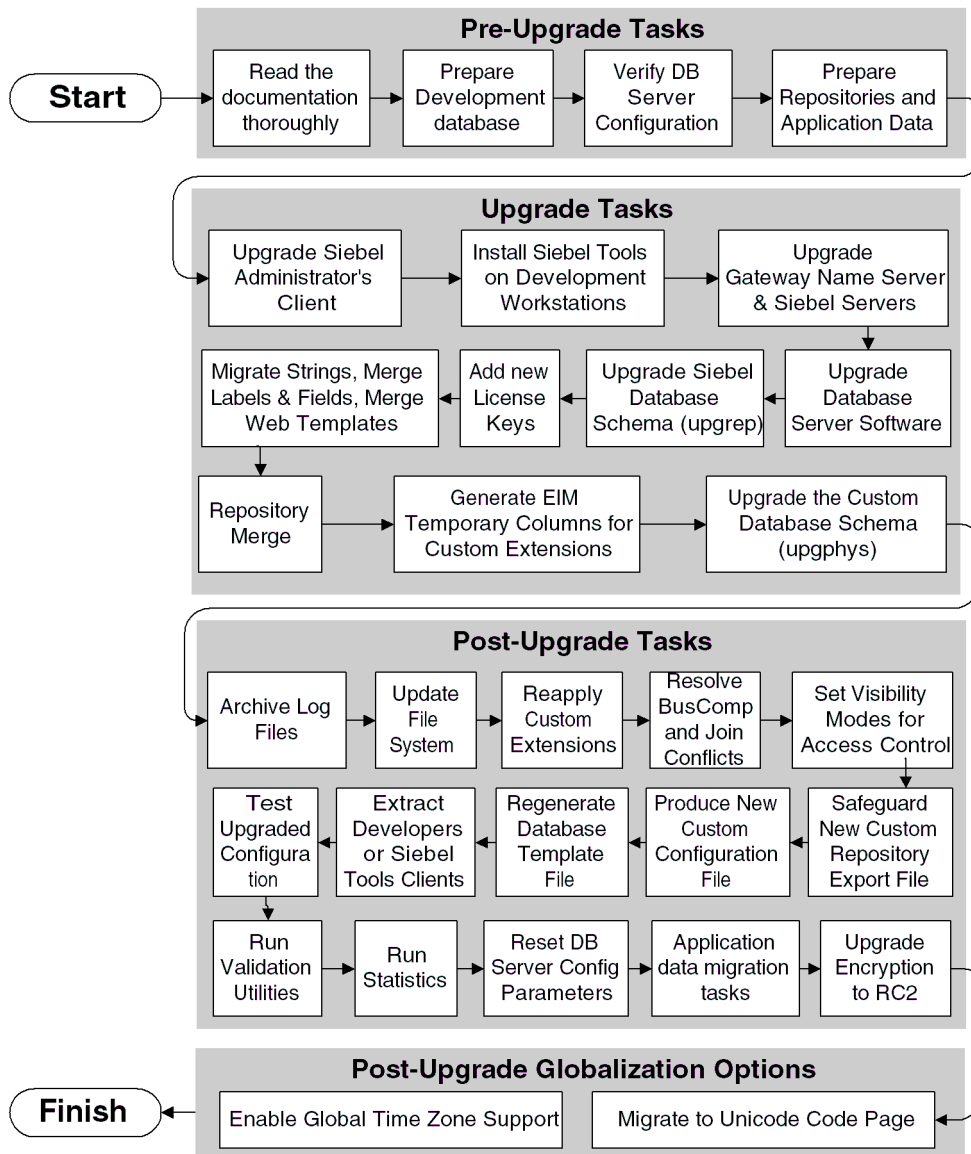


Figure 2. Upgrade of the Development Environment

Planning an Upgrade

Overview of the Upgrade Process

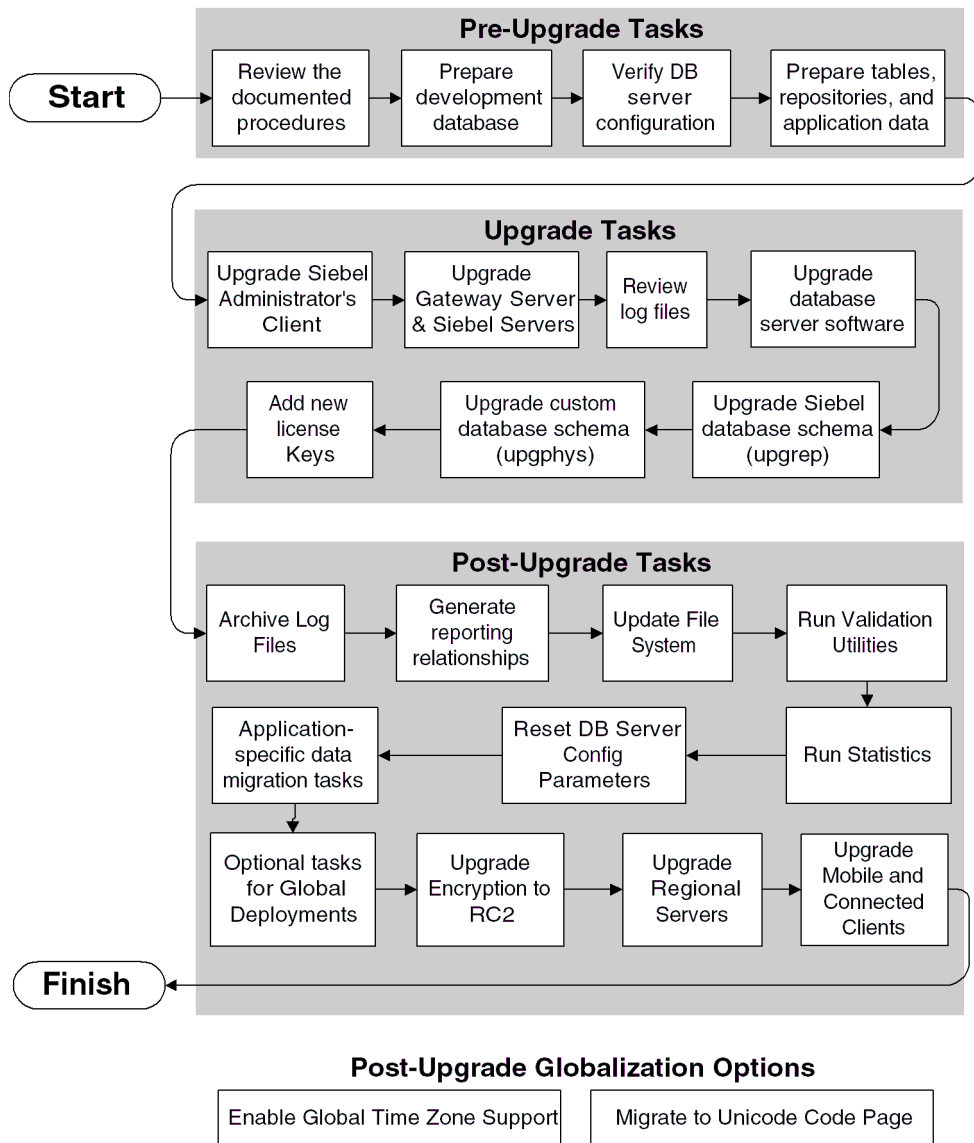


Figure 3. Upgrade of the Production Environment

You will perform the upgrade process in three distinct phases:

- Phase 1 includes upgrading your development environment and performing a repository merge to produce a new Release 7.5 custom configuration. During this phase, your existing configuration and development database are upgraded to Release 7.5.

You must upgrade your development environment before you upgrade your production environment in order to produce and test a Release 7.5 configuration containing your customizations. Then, you must thoroughly test and validate that configuration before you proceed with the production environment upgrade.

NOTE: Users who have not customized Siebel eBusiness Applications, and use Siebel eBusiness Applications as provided by Siebel Systems, Inc., do not have a development environment. If you do not have a development environment, you will skip the development environment phase of the upgrade process.

Follow the instructions in [Chapter 3](#) through [Chapter 5](#) to complete Phase 1.

NOTE: Be sure to test the upgraded custom configuration produced in Phase 1 before proceeding with Phase 2.

- Phase 2 includes migration of your client configuration from Release 6.x or Release 7.0.x Windows client to the Release 7.5 Web client.

Follow the instructions in [Chapter 6, “Migrating the Client Configuration to the Web Client”](#) to complete Phase 2.

NOTE: If you are upgrading from Release 7.0.x to Release 7.5, (for example, from Release 7.0.4 to Release 7.5), you do not need to perform Phase 2.

- Phase 3 includes upgrading your production Siebel Database, Siebel Servers, and File Server or File System. During this phase, you will distribute the new configuration to your production Siebel clients.

Follow the instructions in [Chapter 7](#) through [Chapter 9](#) to complete Phase 3.

Supported Upgrade Paths

This guide does not contain information about supported upgrade paths. For information about supported upgrade paths, including third-party software and hardware, see *Siebel System Requirements and Supported Platforms* on Siebel SupportWeb at <http://ebusiness.siebel.com/supportweb/>.

Nonstandard Upgrades and Migrations

Nonstandard upgrades and migrations are outside the scope of this guide. If you want to perform a nonstandard upgrade or migration, contact Siebel Technical Support or Professional Services for assistance. Nonstandard upgrades and migrations include:

- **Upgrades from one database platform to another.** If you want to change your database platform, contact Siebel Technical Services.
- **Upgrades from one language to another.** Changing the base language of your Siebel application is not supported. However, you can upgrade to the same language version and install the Siebel Language Pack for the desired language. For more information, contact Siebel Technical Services.

Upgrades from Release 7.0.x to Release 7.5

Your upgrade procedure is almost identical to the other supported upgrade paths to Release 7.5, except for several minor variations. Variations are identified at the beginning of each relevant procedure. Carefully read the documentation and perform every procedure in this guide as instructed.

CAUTION: Pay close attention to cautions and notes, for example:

Steps to skip. NOTE: Skip this procedure if you are performing an upgrade from Release 7.0.x to Release 7.5.

Additional steps to perform. CAUTION: If you previously upgraded to Release 7.0.x and you already ran the UTC conversion utility, perform the following steps to upgrade the delta columns (those that were not UTC-enabled in 7.0.x but are UTC-enabled in 7.5).

Steps with replacement steps. NOTE: Skip this step if you are upgrading from Release 7.0.x to Release 7.5, because you already created your 32-KB table space and buffer pool. Instead, perform the task described in “Increasing DB2 32-KB Table Space” on page 154.

The instructions in this guide that are different for your upgrade from Release 7.0.x to Release 7.5 are summarized below.

- You may skip the following procedures, since you already completed them during your upgrade to Release 7.0.x:
 - Data migration procedures in the preupgrade and postupgrade chapters (see [“Preparing Application Data for Upgrade” on page 98](#) for both development and production environments, and [“Additional Postupgrade Tasks for Specific Applications” on page 192](#) for both development and production environments)
 - If your RDBMS is IBM DB2 UDB: Installing stored procedures and user-defined functions (see [“Installing the Stored Procedures and User-Defined Functions” on page 82](#) for both development and production environments)

- If your RDBMS is IBM DB2 UDB: Creating 32-K table space and buffer pool (see [“Creating DB2 32-KB Table Space and Buffer Pool” on page 87](#) for both development and production environments)
- Repository Preparation Wizard to prepare the prior customer repository for the merge (see [“Migrate Strings, Merge Labels and Fields, and Merge Templates” on page 125](#))
- Migrating client configurations to the Siebel Web Client (see [Chapter 6, “Migrating the Client Configuration to the Web Client”](#))
- You may need to perform the following additional procedures:
 - **If your RDBMS is IBM DB2 UDB.** Increasing 32-K table space (see [“Increasing DB2 32-KB Table Space” on page 88](#) for both development and production environments).
 - **If your deployment is enabled for global time zone support.** If you previously upgraded to Release 7.0.x and you already ran the UTC conversion utility, you need to upgrade the delta columns (those that were not UTC-enabled in 7.0.x, but are UTC-enabled in 7.5). See [“Setting Up Your Environment to Support Global Time Zone” on page 202](#).
 - **If you made customizations to Release 7.0.x seeded workflows.** During an upgrade, seeded workflows that were shipped with Release 7.0.x will be replaced by seeded workflows that are shipped with Release 7.5. Customizations will be preserved and migrated, but you will need to manually reimplement them in order for them to work properly after the upgrade. See [“Workflows” on page 41](#).

Migrations to Unicode

To perform a migration to Unicode, contact Siebel Expert Services or Siebel Professional Services for assistance. Migrations to Unicode are outside the scope of this guide.

Before You Start

Before you begin your Siebel upgrade, complete the following steps:

- 1** Gather the username, password, and system information that you will need during the upgrade and complete the Upgrade Planning Worksheet, located in [Appendix A](#).
- 2** Carefully read the chapters in this guide that correspond to your upgrade path to understand the complete upgrade process for your operating system and RDBMS platform combination. Upgrade paths are listed in *Siebel System Requirements and Supported Platforms* on Siebel SupportWeb at <http://ebusiness.siebel.com/supportweb/>.
- 3** Read *Siebel System Requirements and Supported Platforms* to be sure you know the supported computer and operating system platforms and third-party programs for this release of your Siebel eBusiness Applications.
- 4** Schedule database and system administrator assistance.

Upgrading is a database-intensive process. For this reason you should have an experienced database administrator available to assist during your upgrade. Upgrading also requires the installation of new Siebel and third-party software on your Siebel Servers.
- 5** Carefully read the relevant chapters of *Siebel Server Installation Guide* for the operating system you are using.
- 6** Refer to *Release Notes* documentation for new information that this guide does not contain.
- 7** Prepare a comprehensive upgrade schedule that includes a time line for completing specific preupgrade and postupgrade tasks, including thorough testing of your development environment after you upgrade your development environment and before you upgrade your production environment.

Useful Resources

Your Siebel implementation team performs a number of actions to install and implement your Siebel eBusiness Applications that are described in several Siebel publications:

- The current guide for upgrade and configuration instructions for Siebel eBusiness Applications.
- The *Siebel Server Installation Guide* for the operating system you are using for server and RDBMS installation and configuration instructions.
- *Siebel Web Client Administration Guide, MidMarket Edition* for client installation and configuration instructions.
- *Siebel System Requirements and Supported Platforms* available on Siebel SupportWeb at <http://ebusiness.siebel.com/supportweb/> for supported computer and operating system platforms and supported third-party programs for this release of your Siebel eBusiness Applications.
- *Release Notes* documentation for any late-breaking information that the current guide does not yet contain.
- *Siebel Server Administration Guide, MidMarket Edition* for advanced configuration information and details on how to administer, maintain, and expand your Siebel servers.
- If you have a license for Siebel Tools, you will find information about configuring Siebel eBusiness Applications in *Siebel Tools Reference, MidMarket Edition*.
- *Siebel Remote Administration Guide, MidMarket Edition* as a resource for Siebel Remote functionality.

Data Migration Considerations

2

Release 7.x marks a significant improvement and change to existing functionality, enhancing product usability, benefits, and structure of Siebel eBusiness Applications. Those changes have, in turn, required significant changes to the Siebel Data Model. During the Siebel Database upgrade, data held in Siebel Database tables is moved to accommodate this new table structure. This chapter provides information about how and where this data movement occurs so that you can better plan the upgrade of specific modules.

Changes to the Siebel Data Model

This section describes data model changes for Release 7.x including access control, categorization, and the party model. Use this information to plan your upgrade.

Access Control

Access control refers to all mechanisms to control visibility within Siebel eBusiness Applications. This includes but is not limited to positions, responsibilities, organizations, and access groups. Access control is simply creating relationships between people and resources (a more general term for data that includes views and functionality). These relationships or policy are authorizations. Both people and resources can be grouped and placed in hierarchies to ease the administration.

Deployment of Siebel eBusiness Applications over the Web provides external users, such as customers and channel partners with varying access levels, the ability to directly access data and application functionality. This change introduces a new set of data and content access dynamics:

- An exponential increase in the amount of content that will be distributed by the Siebel eBusiness Applications including Master data (data that is static and referential, such as Products) and Customer data (data that is created and managed by users of applications, such as Opportunities).
- An exponential increase in the number of users and entities that will access the data and added complexity of relationships between users (partners, competitors, browsers, customers).
- A significant increase in the complexity of access control policies (one data item or group of data items can be accessed by one or many users or groups, but not by all).

To support access control, data migration occurs during the upgrade to Release 7.5. The following two sections describe this migration.

Multi-Org Visibility for S_CONTACT and S_ORG_GROUP

Before Release 7.5, a Person or Household could not be made visible to multiple Business Units (Organizations). With the introduction of S_CONTACT_BU and S_ORG_GROUP_BU, the application can now share People and Households across multiple Business Units.

The upgrade to 7.5 populates both the S_CONTACT_BU and S_ORG_GROUP_BU tables with one record for each record in the S_CONTACT and S_ORG_GROUP tables. After the upgrade, Contacts and Households continue to be visible from the Business Unit they belonged to prior to the upgrade.

Access Group and Userlist

For 7.5, two new Siebel Extension tables were added to the S_PARTY, S_PARTY_GROUP and S_USERLIST tables to hold Access Group and User List attributes, respectively. The upgrade adds records to the S_PARTY_GROUP and S_USERLIST tables for existing S_PARTY Access Group and User List records. Multi-Org visibility is supported for the new tables, so the upgrade also adds corresponding intersection table records to the S_PARTY_GRP_BU and S_USERLIST_BU tables.

NOTE: For more information about access control, see *Security Guide for Siebel eBusiness Applications, MidMarket Edition*.

Categorization

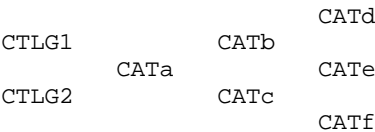
Categorizing content has major benefits: it simplifies access control policy design and management as System Administrators can specify access on an aggregate set of master data items. As a result, content becomes more easily searchable and accessible to users through navigation. Having intuitively navigated to one item, users will likely find many related items of potential interest.

Product categorization was available in the Siebel 6.x data model. However, Release 7.x supports categorization of additional data structures such as literature items.

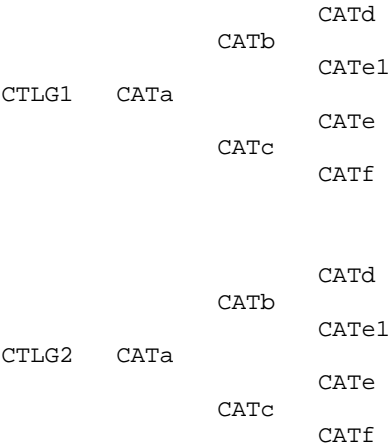
Data Migration to Support Categorization

In Siebel 6.x, categories could be shared across multiple catalogs and could have multiple parents. In Release 7.x, a category can only belong to one catalog and have at most one parent catalog. To accommodate the new structure, the database upgrade generates copies of categories and category hierarchies that were previously shared across multiple catalogs.

For example, the following catalog and category hierarchy:



results in separate hierarchies of copied categories after the upgrade:



NOTE: For more information about categorization, see *Security Guide for Siebel eBusiness Applications, MidMarket Edition*.

Party Model

Release 7.x introduces the concept of a party table (S_PARTY), in which all persons and organizational units are held. With the new model, Accounts, Organizations, Internal Divisions, Contacts, Employees, and Positions are all considered parties and can be referenced from the same table, S_PARTY.

Most of the tables that housed these data entities still exist and are still used, but they are now extension tables to the new base table. Data is brought into the business components through an implicit join.

Additionally, Release 7.x now uses a single-person table and a single-organization unit table. For example, Employees and Contacts are now combined in the same table (S_CONTACT). Similarly, internal and external Organization Units are now combined in the table (S_ORG_EXT).

The new S_PARTY table is the primary table in the Party or Single-Person model and the base table for all Party business components. Siebel extension tables: S_USER, S_EMPLOYEE, S_CONTACT, S_ORG_EXT, S_POSTN, and S_BU support the Party model. Each non-person party directly or indirectly has person members, such as employees or contacts.

The new Party model has made several tables obsolete. The S_EMPLOYEE table is obsolete as its functionality was merged into the S_CONTACT table. The S_ORG_INT table is obsolete, as its functionality has been merged into S_ORG_EXT. S_EMP_POSTN has been replaced by S_PARTY_PER.

There are several other new tables in the data model supporting the consolidation of S_EMPLOYEE with S_CONTACT and of S_ORG_INT with S_ORG_EXT. The S_USER table stores Siebel User information. The S_EMP_PER stores attributes for Brand-Owner Employees and Partner Users who are considered agents of the Brand-Owner, supporting sales and service activities on behalf of the Brand-Owner. The S_BU table stores Organization information.

As a result of the party model, the tables listed in [Table 1](#) are obsolete in Release 7.x:

Table 1. Obsolete Tables

Obsolete Tables in Siebel 7
S_EMPLOYEE
S_EMPLOYEE_ATT
S_ORG_INT
S_ORG_INT_ATT
S_POSTN_RPT_REL

[Figure 4](#) depicts the new Party changes to the data model:

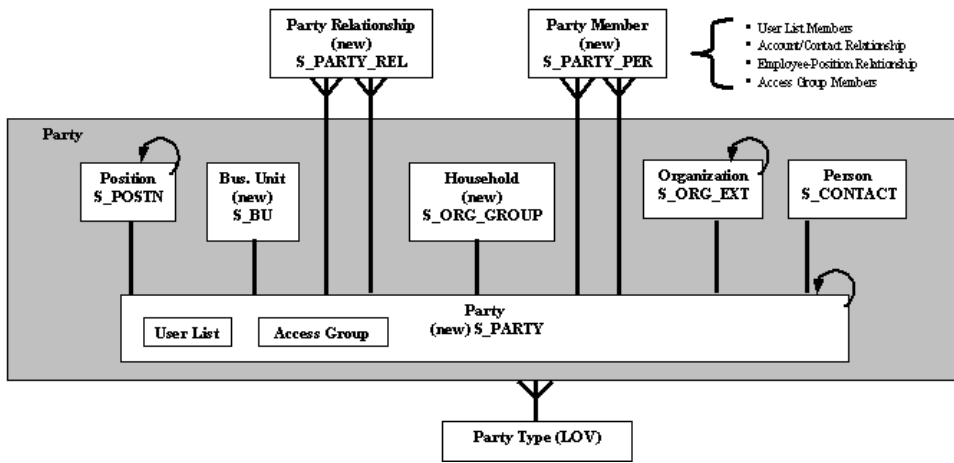


Figure 4. MidMarket Party Model

Given this new table structure and new business object definitions, it is important to understand how the Release 7.x upgrade works. The following steps occur during the Release 7.x upgrade:

Data Migration to Support Party Model

- Migration of data from S_EMPLOYEE to S_CONTACT, S_USER, S_EMP_PER for standard Siebel columns
- Migration of data from S_ORG_INT to S_ORG_EXT, S_BU for standard Siebel columns
- Creation of the Party table records for each previous contact, position, employee, account, division

Business Component Definitions

- Business component definitions updated to reference S_PARTY as Primary Table (for example, Employee, Contact, Position, Account)
- Standard and Custom joins to S_EMPLOYEE changed to S_CONTACT, S_USER, S_EMP_PER
- Standard and Custom joins to S_ORG_INT changed to S_ORG_EXT
- Implicit joins set for custom fields created on business components that have been retargeted to S_PARTY. For example, if a custom field, Alternate Phone, existed on the Contact business component, the upgrade would initiate the following actions:
 - Contact business component would be retargeted to S_PARTY
 - Join to S_CONTACT from S_PARTY defined on Contact business component
 - Implicit join set for the Alternate Phone field

Additional Considerations

With the Access Control mechanism comes two new visibility mode types, catalog visibility and group visibility. For certain standard Siebel eBusiness Application configurations, the view mode and picklist mode have catalog visibility and will only render data visible if the catalogs, categories, and access groups are defined.

As will be discussed later, one of the considerations during the upgrade will be to move forward with the access control model and carefully plan and prepare the access groups, catalogs, categories as part of the upgrade phase.

Alternatively, if continued use of Multi-Org is desired, changes to default view and picklist configuration will need to be modified as part of the upgrade.

NOTE: Make sure data is rendered visible after the production upgrade.

For more details on the Release 7.x data model as it relates to the Party model, please see *Siebel eBusiness Data Model Reference*. For more information on the new Access Control and Security mechanism in Release 7.x, see *Security Guide for Siebel eBusiness Applications, MidMarket Edition*.

Reviewing Columns Added to the S_SRC_PAYMENT Table

If your organization exposes the S_SRC_PAYMENT table to any of its Siebel eBusiness Applications, be aware that upgrading to Release 7.x introduces a new, required column, called TYPE_CD, to this table. This column has a default value of Payment, indicating that this is a payment from your organization to an external organization. For example, this might denote a payment of marketing funds to your customer.

If your organization has used the S_SRC_PAYMENT table to store other types of payments (particularly, payments from your customers to your organization), you will want to reset the parameter in this column to fit your data, and then run Siebel System's Enterprise Integration Management (EIM) product. This will update the table and allow you to review your results before migrating to your production environment.

The valid values for this column as shipped with Siebel eBusiness Applications are:

- Payment
- Receipt
- Write-Off
- Adjustment

For information on how to update the Lists of Values, refer to *Siebel Tools Reference, MidMarket Edition*.

Data Migration Considerations for Release 7.x Applications

This section provides an overview of data migration considerations and functional enhancements in Siebel eBusiness Applications related to their upgrade to Release 7.x. It includes a high-level description of what happens to data during the upgrade to Release 7.x for the following Siebel eBusiness Applications and modules:

- | | |
|-------------------------|------------------------------------|
| ■ Activities | ■ Service |
| ■ Calendar | ■ Campaigns |
| ■ Calendar/Activities | ■ Sales |
| ■ Siebel eChannel | ■ Time Sheet |
| ■ Siebel eMail Response | ■ Universal Time Coordinated (UTC) |

NOTE: Upgrades from Siebel 99.x .COM (HTML Thin Client) applications to Release 7.x eSales, eService, or eChannel require additional steps and custom work. For more information, contact Siebel Technical Support or your sales representative.

Activities

NOTE: The data migration described in this section does not apply to upgrades from Release 7.0.x to Release 7.5 (for example, from Release 7.0.3 or 7.0.4 to Release 7.5).

In Siebel 6.x, users could assign an activity to an individual using the “Assigned To” field. This was a single value field. However, in Release 7.x this behavior has been slightly modified because of the many-to-many relationship that now exists between Employees and Activities, as well as that between Contacts and Activities.

In Release 7.5, the Owner field replaces the Assigned To field. The Owner field is a single value field whose value will be the same as the Assigned To field. Also, the individual that appears in the Owner field will be considered the primary employee for the appointment. The primary employee, as well as all other employees associated with an activity, can be found in the Employee multi-value group (MVG) in both the Activity form and Calendar form applets. Also, employees associated with an activity will now be displayed in the Participant Availability applet in the Calendar and Participant view in the Activities screen.

When activities are migrated from Siebel 6.x to Release 7.x, the value of the Siebel 6.x “Assigned To” field will be moved into the Release 7.x Owner field as well as the Employee MVG and will now be the primary.

Calendar

In previous releases, all calendar appointments appeared in the Activities view by default. During the upgrade to Release 7.x, some activities that appeared in the calendar are migrated to the To Do list. The values in the Suppress Calendar and Type fields of activity records determine where activity records appear after the upgrade to Release 7.x. [Table 2](#) lists the rules that determine the migration of activity records.

Table 2. Activities Migration

Activity records with the following properties:	After the upgrade to 7.x, appear in:
Suppress Calendar = True, Type < > To Do	Activities
Type = To Do	To Do and Activities
Suppress Calendar = False, Type < > To Do	Calendar and Activities

In Release 6.x, appointments were displayed in the Calendar based on the Due field (a date field with no time), the start time (a time field with no date) and the end time (a time field with no date). In Release 7.x, the format of the Start and End fields has been changed such that the fields now display both a date and a time. In addition, appointments are displayed on the Calendar based on the values of the Start and End fields. To account for this change, the following rule is applied during upgrade:

If a 6.x Planned Start is Null, the 6.x Start Time and 6.x Due are merged to create a 7.0 Planned Start or a 7.5 Start time.

This migration is illustrated in [Table 3](#).

Table 3. Calendar Data Migration

Field in 6.x	Field in 7.0.x	Field in 7.5	Upgrade Rule for 6.x to 7.0.x	Upgrade Rule for 6.x to 7.5
Due	Due	Due	No change	No change
Start Time	(obsolete)	(obsolete)	If 6.x Planned Start is Null, then 6.x Start Time + 6.x Due = 7.0 Planned Start	If 6.x Planned Start is Null, then 6.x Start Time + 6.x Due = 7.5 Start
End Time	(obsolete)	(obsolete)	If 6.x Planned Completion is Null, then 6.x End Time + 6.x Due = 7.0 Planned Completion	If 6.x Planned Completion is Null, then 6.x End Time + 6.x Due = 7.5 End
Planned Start	Planned Start	Start	No change	No change
Planned Completion	Planned Completion	End	No change	No change

This logic makes sure that appointments displayed in Siebel 6.x Calendar will also appear in Release 7.x Calendar.

NOTE: In Release 7.0.x, Appointments with a Due Date and no Planned Start appeared on the Calendar. In Release 7.5, this is no longer true. If an appointment has a Due Date, but no Start Date, it will not appear on the Calendar.

Siebel Call Center

During the upgrade of existing Accounts, changes may occur to the status field of Account records. If the status field of an Account record has a value, the value is retained during the upgrade. If the status field has no value, the field is automatically set to “Active” during the upgrade. If the status field has a value of Prospect, that value is updated to Candidate during the upgrade.

Calendar/Activities

NOTE: The data migration described in this section does not apply to upgrades from Release 7.0.x to Release 7.5 (for example, from Release 7.0.3 or 7.0.4 to Release 7.5).

- View calendars of multiple users simultaneously; see free and busy time for each.
- Add multiple employees and contacts to the same activity.
- Create queries in Activity Views and apply them to Calendar Views.
- Use Calendar as a child applet in noncalendar screens such as Accounts and Contacts.
- Tools tips (Hover over Activity for more information).
- Coordinated Universal Time Support.
- All activities with either a Planned Start or Due value will be displayed on the calendar.

Siebel eChannel

NOTE: The data migration described in this section does not apply to upgrades from Release 7.0.x to Release 7.5 (for example, from Release 7.0.3 or 7.0.4 to Release 7.5).

In Release 7.x, the new Party model merges all person and party tables, for example, Contact, Account, Internal Division, Position, into one table called S_PARTY. This new Party schema provides a range of benefits in terms of how person and party entities, and the relationships between entities, are defined. Entity-specific data, such as information specifically about contacts or accounts, will still be stored in an S_PARTY extension table, such as S_CONTACT and S_ORG_EXT.

Prior to Release 7.x, eChannel customers modeled their partners as both Accounts (in S_ORG_EXT) and Divisions or Organizations (in S_ORG_EXT). Similarly, partner employees were modeled as both Contacts (in S_CONTACT) and Employees (S_EMPLOYEE). In Release 7.x, this duplication has been eliminated. Partner companies and their employees will have one record each in S_PARTY, with company or employee specific data in the corresponding extension table.

While the new Party model will have a significant impact on how the data appears in the database, it will not significantly change how you view this data in the Siebel eChannel Partner Manager application.

Another significant functional enhancement in Release 7.x is access control for master data (data that is static and controlled by Siebel administrators). This feature provides the ability to create hierarchical catalogs of master data and hierarchical user (or access) groups. Access to the catalogs can be provided at any level of the hierarchy to any number of access groups. Catalogs can also be accessible to all users. This feature will make the administration of master data, such as literature, solutions, and products much easier for companies who are managing multiple partner organizations.

eChannel customers who have been using multiple organization visibility to master data should carefully assess how they would like to view data going forward. Siebel eChannel Partner Portal uses the new access control functionality for visibility to master data in all associated list views or popups. This data will not be affected by the upgrade, but customers should either migrate to access control or reconfigure the application to maintain multiple organization visibility. If nothing is done, customers who currently use multiple organization visibility will lose visibility to their master data. Refer to the Access Control Upgrade Migration technical note on Siebel SupportWeb for more information about migrating to access control.

In Release 7.x, the product catalog uses the access control mechanism for categorization and visibility to buying catalogs.

Siebel eMail Response

NOTE: The data migration described in this section does not apply to upgrades from Release 7.0.x to Release 7.5 (for example, from Release 7.0.3 or 7.0.4 to Release 7.5).

The Contact Email Address field has been changed in Release 7.x to only store lowercase values. This change was made to address a performance issue with Siebel eMail Response. If you want to disable this behavior, you can use Siebel Tools to change the Force Case property on the Email Address field in all business component fields that reference the S_CONTACT.EMAIL_ADDR column.

Existing records in the Siebel eBusiness Applications database are not converted to lowercase during the upgrade process. If you are using Siebel eMail Response, you may want to run a database script to convert the existing records to lower case. This will ensure that the contact lookup functionality will accurately identify contacts.

Expense Reports—Hotel Lines

In Release 7.5, Expense Reports have been enhanced to improve usability and decrease the number of clicks and drilldowns required to create an expense report. In pre-7.5 releases, hotel expenses could be captured with one expense line item while the details about individual nights and/or charges could be captured as subitems by drilling down on the line item. In Release 7.5, each hotel night is captured as an individual expense line item. To support this new paradigm, pre-7.5 hotel subitems are upgraded to expense line items, using the following rules:

- All fields are copied from original line item to the new subitem, excluding EXP_ITEM_NUM, START_DT, END_DT, DESC_TEXT, PAR_EXP_ITEM_ID, and all fields ending with _AMT.
- Total Amount for new expense line should be the sum of Room Amount, Room Tax Amount, Room Other Tax Amount, Breakfast Amount, Lunch Amount, Dinner Amount, Business Phone Amount, and Misc. Amount of the original subitem (does not include Personal Phone Amount).
- Personal Phone Amount is handled according to the following rules:

If any of the subitems on a hotel record have a Personal Phone Amount > 0, the Total Amount of the original expense line will equal the sum of all subitem Personal Phone Amounts. In addition the following values will be updated to the original expense line; Reimbursable = N, Type = Personal.

If no sub-items contain Personal Phone Amounts, original expense line Total Amount is set to 0, and the following comment is appended to the first 100 characters of the existing comment - “This record is intentionally set to 0 during the upgrade process.”

NOTE: To maintain the integrity of already billed items, hotel expense items where the BILLING_DT is not NULL are not upgraded.

Workflows

NOTE: The data migration described in this section *only* applies to upgrades from Release 7.0.x to Release 7.5 (for example, from Release 7.0.3 or 7.0.4 to Release 7.5).

Skip this section if you are upgrading from Release 5.x or Release 6.x.

During an upgrade, seeded workflows that were shipped with Release 7.0.x (for example, Release 7.0.3 or 7.0.4) will be replaced by seeded workflows that are shipped with Release 7.5. Your customizations will be preserved and migrated, but you will need to manually reimplement them in order for them to work properly after the upgrade.

- **Seeded workflows.** Seeded workflows have inactive status and version 0. They cannot be modified.
 - To use a seeded workflow, revise and activate the workflow so that a new copy of the seeded process will be created with the latest version number.
 - To maintain your customization from a previous release but incorporate the changes in the new seeded workflows, you need to perform a manual three-way merge. That is, you need to revise the seeded workflow (that is shipped as version 0), manually merge in your customizations, and then activate the workflow.
- **Nonseeded workflows.** Nonseeded workflows will be migrated as such, carrying over respective version numbers.

Service

NOTE: The data migration described in this section does not apply to upgrades from Release 7.0.x to Release 7.5 (for example, from Release 7.0.3 or 7.0.4 to Release 7.5).

- In Orders, Sales Team visibility has been added. Consequently, for users to see existing Orders, they must be added to the Sales Team MVG.
- On Agreements, Sales Team visibility has been added.
- Assets has Sales Team visibility and is a single Organization. For this reason, the appropriate Organization should be stamped on existing Assets.

Siebel Campaigns

Position-Based Campaign Contacts

In Release 7.5, there is a new position-based field for campaign contacts. This field is populated with an employee's primary position referenced by the campaign contact's ownership based field. This position-based field is based on the column S_CAMP_CON.POSTN_ID, which is an existing column in Release 7.0.x that was never exposed. This field is now used as the Position Owner for a Campaign Contact. During the upgrade, it is overwritten with the Primary Position of an employee.

Many-to-Many Relationship Between Campaigns and Contacts

In Release 7.0.x and prior releases, the combination of CON_PER_ID and SRC_ID in S_CAMP_CON is unique. In Release 7.5 this uniqueness is expanded to include the functionality of contacting the same contact multiple times under the same campaign, but in different contexts.

In Release 7.0.x, the Analytics module used to figure out the segment information from S_CAMP_CON based on the contact and campaign information. With the relaxation of this uniqueness, this is no longer possible. As such, new CAMP_CON_IDs are added to the S_OPTY, S_ORDER and S_COMMUNICATION tables in Release 7.5 and are populated during the upgrade with the correct campaign contacts.

Multi-Org Visibility for S_SRC and S_CALL_LST

In Release 7.5, new functionality was introduced to support multi-org visibility for programs, campaigns and segments. During the upgrade, the new tables S_SRC_BU and S_CALL_LST_BU are populated with records from S_SRC and S_CALL_LST respectively.

NOTE: The data migration described in this section does not apply to upgrades from Release 7.0.x to Release 7.5 (for example, from Release 7.0.3 or 7.0.4 to Release 7.5).

For 7.x, a data upgrade issue for Siebel Campaigns concerns external data mapping views. For example, if marketing application administrators have mapped to a table in the Siebel OLTP or OLAP where the schema has been changed for Release 7.x, they have to manually update their data mapping. Data migration that occurs during the upgrade process is described below.

- For any Campaign with no defined primary team member, the sadmin record is added as the primary team member.
- Default Campaign Name and Default Offer Name are updated in System Preferences as Default Campaign Source Code and Default Offer Code, respectively. When the upgrade process is completed, system administrators need to change the System Preferences Value from Campaign and Offers Names to their source codes. If they do not make this change, Siebel Campaigns will not work properly. This change was made due to the fact that “SRC_NAME” is no longer unique, but Source Code is. (In the event of globalization, these might be moved to the .cfg files.)
- During the upgrade, the Call Guide is now copied into S_NOTE_DCP when the Offer is generated. In Release 7.x contents of Call Guide attached to a campaign have moved to an Offer of type “Phone.” As a result, if such campaigns did not have a Phone offer attached to them, a dummy offer will be created with this call guide content and attached to this campaign. If the campaign has one or more phone offers, then this call guide is to be attached to the primary phone offer.

Orders

In Release 7.5 of Siebel Orders, multiple methods of payment, (credit card, purchase order, and so on), can be used to submit payments. To support this new functionality, during the upgrade to Release 7.5, data from fields in the Order Entry - Orders business component is copied to fields in the Payments business component. This migration is detailed in [Table 4](#).

Table 4. Orders Data Migration

Fields in Order Entry - Order Business Component	Fields in Payments Business Component
Bill To Contact Id	Payment Contact Id
Bill To Account Id	Bank Account Id
Bill To Account Id	Bill To Address Id
Bill To Account Id	Payment Account Id
Payment Method	Payment Method
Credit Card Name	Payment Type
Credit Card Approval Code	Authorization Code
Credit Card Expiration Date	Expiration Date
Credit Card Expiration Month	Expiration Month
Credit Card Expiration Year	Expiration Year
Credit Card Holder	Card Holder
Credit Card Merchant Id	Merchant Id
Credit Card Number	Account Number
Credit Card Number Key Index	Encryption Key
Credit Card Transaction Amount	Transaction Amount
Credit Card Transaction ID	Transaction Id
Credit Card Transaction Response Code	Transaction Message
Credit Card Transaction Status	Transaction Type
Credit Card Transaction Time	Transaction Time

Table 4. Orders Data Migration

Fields in Order Entry - Order Business Component	Fields in Payments Business Component
Currency Code	Currency Code
Exchange Date	Exchange Date
Account Order Number	Account Number
Personal Bill To City	Personal Bill To City
Personal Bill To Country	Personal Bill To Country
Personal Bill To Postal Code	Personal Bill To Postal Code
Personal Bill To State	Personal Bill To State
Personal Bill To Street Address	Personal Bill To Street Address
Personal Bill To Street Address 2	Personal Bill To Street Address 2

In addition, in Release 7.0.x, the `Payment Method` field in the Order Entry - Orders business component was not mandatory. In Release 7.5, this field is required in the Payments business component. During the upgrade to Release 7.5, this field is populated in the Payments business component according to the following rules:

- If the field `Account Order Number` is populated in the Order Entry - Orders business component, the `Payment Method` field in the Payments business component is populated with Purchase Order from the LOV `PAYMENT_METHOD_CODE`.
- If the field `Credit Card Number` is populated in the Order Entry - Orders business component, the `Payment Method` field in the Payments business component is populated with Credit Card from the LOV `PAYMENT_METHOD_CODE`.

Siebel Professional Services Automation (Siebel PSA)

This section describes data migration to support Siebel Professional Services Automation 7.5.

Migrating Access Type of Projects from Private to Public

In 7.5, Siebel PSA introduces a new access type field for projects. This allows projects to be designated as private, public, or public read only.

- **Private.** User must be on access list of project for read/write visibility.
- **Public.** All users have read/write access to project.
- **Public Read Only.** Users on access list have read/write access, all other users have read access.

If no access type is selected, the behavior is private. In the upgrade, all items will have no access type, so they will act like private projects.

Your organization may want to migrate common projects (for example, Vacation, Sick, or Holiday) to public projects. To do this, change the access type on the project to `public`.

Upgrade of Project Activities

During the upgrade to Release 7.5, project activities that have Parent Tasks are migrated to Project Tasks. This migration occurs in support of the enhanced Microsoft Project Integration offered in Release 7.5. The Project Activities are not deleted after the migration has occurred. In order to clean up the work breakdown structure of the project, as well as the work assignments for your employees, (following the migration, employees will have both an Activity and Task assigned to them representing the same work item), you may want to consider deleting these Project Activities after the upgrade.

NOTE: Not all fields are migrated from Project Activity to Project Task, for example, Activity Comments is not migrated. For this reason, you must carefully review all fields that are migrated. If you find that fields you need are not migrated to Project Activity, you may decide not to delete the original Project Activities after the upgrade.

You can use the following SQL statement against the post-upgrade database to delete the original Project Activities:

```
delete from S_EVT_ACT
where PROJ_ITEM_ID = ROW_ID
and ASSOCIATED_COST = 0
and EVT_STAT_CD = (select lv.VAL
                    from S_LST_OF_VAL lv
                    where lv.NAME = 'Unscheduled'
                    and lv.TYPE = 'EVENT_STATUS'
                    and lv.LANG_ID = UPPER('&Arg2'))
)
and exists
(select 'x'
 from S_PROJITEM pi
,S_LST_OF_VAL lov
where S_EVT_ACT.PROJ_ITEM_ID = pi.ROW_ID
and S_EVT_ACT.ROW_ID = pi.ROW_ID
and pi.TYPE_CD = lov.VAL
and lov.NAME = 'Requirement'
and lov.TYPE = 'PSTASK_TYPE'
and lov.LANG_ID = UPPER('&Arg2'))
)
/
```

The migration from Project Activities to Project Tasks is done by copying the data from each Activity to a new Project Task, using the field mappings and rules in [Table 5](#).

Table 5. Field Mappings for Project Activities and Project Tasks Migration

Field Mappings and Rules
Task ID = Activity ID
Task Name = Activity Description
Task Type = Activity Type
Task Planned Start = Activity Planned Start

Table 5. Field Mappings for Project Activities and Project Tasks Migration

Field Mappings and Rules
Task Planned Completion = Activity Planned Completion
Task Status = Activity Status
Task Priority = Activity Priority
Task Assigned To = Activity Employees
Task Due = Activity Due
Task % Complete = Activity Complete %
Task Project = Activity Project
Task Actual Start = Activity Actual Start
Task Actual Completion = Activity Actual Completion
Task Effort Remaining = Activity Effort Remaining
Task Duration (Hours) = Activity Duration (Minutes)
Task Cost = Activity Cost Estimate
Task Parent Task = Activity Parent Task
Task Created = Activity Created
Task Created By = Activity Created By
Task Description = Activity Comments
Task Primary Requesting Customer = Activity Account
Task Parent Task = Activity Parent Task

Additional Rules

- Activity Team is migrated to Task Team (including Primary status), potentially requiring the addition of Team Workbook records, as follows:
 - If the employee is already assigned to one role in the Team Workbook, Task assignment will use the existing Team Workbook record.
 - If the employee is already assigned to multiple roles in the Team Workbook, a new Team Workbook record will be created for this employee and Task assignment will use this new Team Workbook record. A new one will be created for each instance of this occurrence.
 - If the employee is not currently assigned to a role in the Team Workbook, a new Team Workbook record will be created for this employee and Task assignment will use this new Team Workbook record. After the first new record is created, it will be used for each instance.
- Attachments are copied from original Activity to Task.
- Time, Expense, and Invoice items are associated to a new Task.

NOTE: The association with the Project Activity still exists. If the Project Activities are being deleted, as stated above, you may want to consider clearing the Activity fields for these items as well.

- For Original activities of Type = Milestone, Milestone flag is set on Task.
- Updates to original Activities upon completion (the Project Activity is not deleted):
 - Status set to Unscheduled
 - Cost set to 0 (to avoid double counting in the budget)
 - Activity Parent Task set to newly created Task

Activities without Parent Tasks are not upgraded in this manner. Thus, level 1 leaf nodes originally imported from Microsoft Project will not be migrated. If you imported a Microsoft Project file in Release 7.0.x, upgrade to Release 7.5, then re-import the same file, an additional task will be created. As a result, you may want to delete the original activity.

Upgrading Siebel Quotes with Customizable Products and Service Products

NOTE: The data migration described in this section does not apply to upgrades from Release 7.0.x to Release 7.5 (for example, from Release 7.0.3 or 7.0.4 to Release 7.5).

To make sure quotes that use products as solutions in Siebel 6.x will be upgraded to Release 7.x, verify that the “Model Product” field is unchecked.

In Siebel Quotes 6.x, the Discount Amount field on a line item was automatically populated to “\$0.00.” This meant that, for any line item, users had to clear the discount amount field if they wanted to apply a header level discount after adding a line item.

However, some customers implemented Siebel Quotes 6.x to read the Discount Amount field the same whether it was “NULL” or “\$0.00,” thereby allowing users to enter a header discount after adding a line item and avoiding the situation illustrated above.

In Release 7.x, this issue is no longer applicable. The Discount Amount field is left as “NULL” unless you explicitly enter a value, including “\$0.00.” Therefore, when you upgrade from Siebel 6.x to Release 7.x, you need to determine whether a change was made in Siebel Quotes 6.x to Discount Amount (therefore letting it be automatically populated to “\$0.00”) or whether Siebel Quotes 6.x has been modified to read “NULL” or “\$0.00.” In the event the latter is true, you must run a Quote script before beginning the upgrade, that is, before running the configuration utility that is applied to development and production upgrades. The Quote script must be preceded by the SQL statements listed below.

NOTE: These two statements must be run before the upgrade. Run the first statement only if a discount value of “0” (zero) is treated as empty. You will run the second statement after the first one, if it is run. Run the second script regardless of whether the first is run or not.

- **Statement 1.** Run this statement only if a discount amount of “0”(zero) is treated as empty.

```
update    S_QUOTE_ITEM
set       DISCNT_AMT = NULL
where     DISCNT_AMT = 0
;
commit
;
```

This script will account for the fact that the Discount Amount field is read the same whether it is “NULL” or “\$0.00.” Administrators running Siebel Quotes 6.x without modifications to Discount Amount can ignore this script.

- **Statement 2.** Run this statement regardless of whether the previous script is run or not, but if you run the previous script, run this one afterwards.

```
update    S_QUOTE_ITEM
set       DISCNT_AMT = 0.0
where     UNIT_PRI is NULL
and       DISCNT_PERCENT is NULL
and       DISCNT_AMT is NULL
and       exists
(
    select 'x'
    from    S_PROD_INT
    where   ROW_ID = S_QUOTE_ITEM.PROD_ID
    and     SALES_SRVC_FLG = 'Y'
)
;
```

This script will make sure that all service products do not take any header level discounts.

- After upgrade, you need to run a statement to remove unnecessary trailing spaces from quote items. This is a mandatory postupgrade step; see [“Quotes” on page 193](#) for development environment upgrades or [“Quotes” on page 329](#) for production environment upgrades.

Siebel Sales

After an upgrade from Release 7.0.x to Release 7.5, if you use the Target Account Selling (TAS) proposal generator, you need to manually copy the following TAS Opportunity Plan template files from the seed database to the file system:

S_DOC_PPSL_0-7VC8N_0-2AQ.SAF
S_LIT_0-7VCBO_0-2AR.SAF
S_LIT_0-7VCBQ_0-2AS.SAF
S_LIT_0-7VCBS_0-2AT.SAF
S_LIT_0-7VCBU_0-2AU.SAF
S_LIT_0-7VCBW_0-2AV.SAF
S_LIT_0-7VCBY_0-2AW.SAF
S_LIT_0-7VCC0_0-2AX.SAF
S_LIT_0-7VCC2_0-2AY.SAF
S_LIT_0-7VCC4_0-2AZ.SAF
S_LIT_0-7VCC6_0-2B0.SAF
S_LIT_0-7VCC8_0-2B1.SAF
S_LIT_0-7VCCA_0-2B2.SAF
S_LIT_0-7VCCC_0-2B3.SAF
S_LIT_0-7VCCE_0-2B4.SAF

NOTE: The data migration described below does not apply to upgrades from Release 7.0.x to Release 7.5 (for example, from Release 7.0.3 or 7.0.4 to Release 7.5).

Opportunities no longer support the following fields in the Sales Team multi-value group (MVG):

- Close Date
- Probability
- Committed
- Position Territory

There are changes in the values of the Revenue Class field and activity types. Also, for Target Account Selling (TAS), the views and fields listed in [Table 6](#) were deleted.

Table 6. Views Deleted for Target Account Selling (TAS)

Deleted Views
Contacts - TAS - Profile
Contacts - TAS - Behavior
Contacts - TAS - Background
Accounts - TAS - Products
Accounts - TAS - Business Description
Accounts - TAS - Trends and Objectives

Other changes for TAS include the following:

- The Opportunity - TAS - Overview Account Profile field used to pull data from the account and is now no longer linked to the Account Profile but is able to be edited in the Opportunities screen.
- Milestones used to be a list applet in the Opportunity - TAS - Overview view. Now it has its own view.
- Shared Notes used to be a list applet in the Opportunity - TAS - Assessment view, but now it has its own view.
- Users had to toggle between the Assessment form applet and the Assessment list applet in the Opportunity - TAS - Assessment view. Now these are separate applets.
- The Opportunity - TAS - Solution view no longer exists; the Products list applet is no longer supported within the module; and the Unique Business Value, Strengths and Weaknesses fields have been moved to the Opportunity - TAS - Overview view.
- The Order Amount and Close Date fields are no longer supported in the Opportunity - TAS - Competitive Analysis view.

- The Anticipated Strategy values have been changed in the Opportunity - TAS - Competitive Analysis view.

Old values were:

- | | |
|------------|------------|
| ■ Defend | ■ Fragment |
| ■ Develop | ■ Frontal |
| ■ Flanking | |

New values are:

- | | |
|---------------------------------|---------------------------------|
| ■ Defend-Insulate | ■ Flanking-Alter the Rules |
| ■ Defend-Isolate | ■ Fragment-Niche |
| ■ Develop-Delay | ■ Fragment-Peaceful Coexistence |
| ■ Develop-Invest | ■ Frontal-Reputation |
| ■ Flanking-Acknowledge & Expand | ■ Frontal-Solution |
- The Tactics list applet in the Opportunity - TAS - Plan view now has its own view.
 - The Goal, Critical Success Factors and Solution fields are now supported in the Opportunity - TAS Overview view instead of the Opportunity - TAS - Plan view.
 - The Opportunity - TAS - Plan view is no longer supported.

In Siebel Assistant Administration, Views have been moved to beneath Application Administration, and the Siebel Assistant Administration category has been removed.

- Opportunity revenue amounts, win probability, commit status, and close date has been moved from oppty_postn to the revenues table. See *Applications Administration Guide, MidMarket Edition*.
- Activity date time changes: UTC and migration of a few activity date/time fields.
- Activity contact MVG, activity employees MVG.
- Accounts and contacts have been moved into the new Party table.
- Employees and Divisions have been moved into the new Party table.

Time Sheet

If you are upgrading from a pre-7.0.x version of Siebel eBusiness Applications to Release 7.5, please refer to Siebel SupportWeb for a technical note about Time Sheet Conversion.

Universal Time Coordinated (UTC)

In Siebel 6.x any references to time were based on the clock running on the local machine where the record originated. In Release 7.x, time is independent of the local machine's clock. Instead, all references to time are stored in the Release 7.x database based on Greenwich Mean Time (GMT). Users have the ability to specify a preferred time zone in which they would like to view all references to time. In order to migrate references to time from Siebel 6.x to Release 7.x, logic has been developed which determines the time zone in which a Siebel 6.x time value was created. Based on this time zone, the time value is converted to GMT and moved to the Release 7.x database. Finally, when a Release 7.x user retrieves a record with a time value, that time value is rendered based on the user's preferred time zone. See *Global Deployment Guide, MidMarket Edition* for more detailed information on this subject.

NOTE: If you previously upgraded to Release 7.0.x (for example, Release 7.0.3 or Release 7.0.4) and you already ran the UTC conversion utility, you will need to perform a postupgrade procedure to upgrade data from columns that were *not* UTC-enabled in 7.0.x that are UTC-enabled in 7.5. See [“To upgrade 7.0.x data for columns that have changed to UTC-enabled in 7.5.”](#)

Development Environment Preupgrade Tasks

3

This chapter describes the tasks that must be performed before you upgrade your development environment, the first step in upgrading your deployment. A checklist of preupgrade tasks is provided in [Table 7 on page 59](#).

NOTE: If you have made no customizations to your Siebel eBusiness Applications, and therefore do not have a development environment, proceed directly to [Chapter 7, “Production Environment Preupgrade Tasks.”](#)

Before You Begin

If you have not already done so, copy the Upgrade Planning Worksheet, located in [Appendix A, “Upgrade Planning Worksheet,”](#) and fill out the appropriate fields with the following:

- **Siebel Administrator User Name and Password.** This must be a valid RDBMS user name and password, and it must be set up as a Siebel employee. The employee record must have the “Siebel Administrator” responsibility. `SADMIN` is the default administrator user name and password. If this user does not already exist in your database, or does not have “Siebel Administrator” privileges, then you must add this to your database prior to proceeding with the upgrade.
- **Database Tableowner Account User Name and Password.** For DB2 UDB, Microsoft SQL Server, and Oracle, `SIEBEL` is the default tableowner account user name and password for Siebel eBusiness Applications.
- **Siebel Data Table Space.** Only applicable to Oracle, this is the name of the table space on the Oracle server where the Siebel data tables are stored.
- **Siebel Index Table Space.** The name of the table space on the DB2 or Oracle server where the Siebel indexes are stored.
- **Siebel 4-KB Data Table Space.** The name of the table space on the DB2 server where the 4-KB Siebel data tables are stored.
- **Siebel 16-KB Data Table Space.** The name of the table space on the DB2 server where tables reside whose row length equals greater than 4005 bytes, but less than 16,384 bytes.
- **Siebel 32-KB Data Table Space.** The name of the table space on the DB2 server where tables reside whose row length equals up to 32,768 bytes.

NOTE: Siebel table spaces for the DB2 UDB platform should be database managed table spaces (DMS) rather than system managed table spaces (SMS).

A checklist of preupgrade tasks is provided in [Table 7](#).

Table 7. Siebel Development Environment Preupgrade Tasks

	Preupgrade Task
1	Fill out your copy of Appendix A, “Upgrade Planning Worksheet” with information specific to your RDBMS platform.
2	Verify database sort order. See “Verifying Database Sort Order” on page 60 .
3	Save all pending changes. See “Saving All Pending Changes” on page 63 .
4	Back up the development database. See “Backing Up the Development Database” on page 63 .
5	Save interface table data. See “Saving Interface Table Data” on page 64 .
6	Upgrade your RDBMS software. See “Upgrading Your RDBMS Software” on page 66 .
7	Verify database server configuration for your RDBMS. See “Verifying Database Server Configuration” on page 68 .
8	Preserve custom indexes on tables. See “Preserving Custom Indexes on Tables” on page 92 .
9	Disable customized triggers. See “Disabling Customized Triggers” on page 93 .
10	Drop customized views. See “Dropping Customized Views” on page 93 .
11	Rename the existing development repository from “Siebel Repository” to “Prior Customer Repository”. See “Renaming Your Existing Development Repository” on page 96 .
12	Record dock objects and visibility rules. See “Recording Dock Objects and Visibility Rules” on page 97 .
13	Prepare your application data. See “Preparing Application Data for Upgrade” on page 98 .

Preparing the Development Environment Database for Upgrade

Before you upgrade your development environment, make sure that the development database configuration meets the database requirements outlined in [“Verifying Database Server Configuration” on page 68](#), and meets the requirements depicted in *Siebel Server Installation Guide for Microsoft Windows, MidMarket Edition*.

Verifying Database Sort Order

Verify that your development database was created with binary (identity for IBM DB2 UDB) sort order.

CAUTION: Development databases *must* use binary sort order to prevent an error from occurring during the repository merge. If your local language does not use binary sort order, you need to recreate your database to use binary sort order.

If you need more information about how to set the order in which your database sorts data, see *Siebel Server Installation Guide for Microsoft Windows, MidMarket Edition*. If you need assistance on setting these parameters, contact your DBA.

Verifying Sort Order on Microsoft SQL Server

Siebel Systems recommends that you set your sort order to *binary* sort at the instance level at the time of your installation or upgrade of SQL server (the default installation setting is typically dictionary). Each database created under this instance that will be used by the Siebel application should be set to binary sort order. Consult your vendor documentation to verify sort order at the instance and database levels.

CAUTION: For development environments, you must set sort order to binary. This is required for repository merge to function correctly; otherwise, it may result in repository data corruption.

To verify that your database was created using binary sort order

- 1 In the Query Analyzer window, enter the following command:

```
sp_helpsort
```

This command will provide a sort order description.

- 2 Review the sort order description to verify binary sort order; for example,

```
Binary Sort Order for the ISO 8859-1 (Latin 1) Character Set
```

NOTE: If you find that your Microsoft SQL Server database was not created using binary sort order, you must rebuild your database and reload your data. Please review Microsoft documentation for detailed instructions.

Verifying Sort Order on Oracle Client

Siebel eBusiness Applications only supports binary sorting under Oracle. Sort order is determined by the NLS_SORT parameter on the Oracle Client. You must set NLS_SORT to BINARY, or you must choose a NLS_LANG setting that includes binary.

CAUTION: The NLS_LANG parameter must be set to the same value throughout your enterprise, and it must match the database character set. The NLS_LANG parameter is required for conversion from a non-Unicode code page to Unicode; an incorrect setting could lead to data loss.

To verify that your database was created using binary sort order

- 1 Use sqlplus to connect the the Oracle database.

- 2 Issue the following query:

```
SQL> SELECT * FROM NLS_DATABASE_PARAMETERS;
```

- 3 Review the returned parameters for NLS_SORT, and verify that the value for this parameter is BINARY.
 - If NLS_SORT has a value of BINARY, then the default sort order is binary and no action is required.
 - If NLS_SORT is anything other than BINARY, then you need to recreate the database so that it uses binary sort order.

Verifying Sort Order on IBM DB2 UDB

Siebel Systems supports only identity sort order on your DB2 UDB database. Perform the following steps to verify the sort order on IBM DB2 UDB.

To verify that your database was created using identity sort order

- 1 Run the following query on Siebel database:

```
select count (*) from S_APP_VER where '$' > '/'
```

- 2 Review the result.

- If sort order is correct, the result will be

```
1
-----
0
(1) record selected.
```

- If sort order is incorrect, you need to recreate the database, using the option:

```
COLLATE USING IDENTITY
```

NOTE: Sort order is specified during creation of the database. If you find that your IBM DB2 UDB database was not created using identity sort order, you must recreate your database using the option `COLLATE USING IDENTITY`.

If sort order is correct, but you are still encountering errors, please contact Siebel Technical Services for further analysis.

Saving All Pending Changes

Developers who are using Siebel Tools with a local SQL Anywhere database must check in their projects to the development repository. Make sure that all project locks in your current Siebel repository have been released to prevent inadvertent loss of development work during the upgrade.

If you are using Siebel Workflow Manager, you must run the Workflow Monitor Agent and Workflow Action Agent to completion before upgrading to Release 7.x. The `S_ESCL_REQ` table should not have any rows if Workflow Manager has completed successfully.

CAUTION: Before starting the upgrade, all locked projects must be checked in and unlocked. If projects are not checked in and unlocked, errors may occur during the repository merge process.

Backing Up the Development Database

Perform a full backup of the development database, using the utilities specific to your database platform. This backup protects your development repositories and environment.

Siebel Systems recommends that you back up your database repository at key stages of the development environment upgrade:

- Before any upgrade activity is started
- After upgrading the Siebel Database Schema
- After the repository merge
- After upgrading the Custom Database Schema

Saving Interface Table Data

Use the appropriate tools for your RDBMS to export any data in your interface tables that you want to retain. During the upgrade process, your interface tables will be dropped and then recreated. After you have upgraded, you can then import this data using the tools for your RDBMS.

NOTE: During the upgrade, all custom indexes on interface tables are dropped from both logical and physical schema.

Identifying and Resolving Duplicates

Before you upgrade to Release 7.5, you need to resolve any duplicates in your Siebel database. To identify duplicates, run the script, `Find_DUP_S_PARTY_ROW_IDS.sql`, located in the `DBSRVR_ROOT\DatabasePlatform\upgrade` directory.

This SQL script will generate a list of duplicates (if any) that you must resolve prior to beginning the upgrade to Siebel 7.5.

Preparing Mobile and Dedicated Users for the Upgrade

Complete the following actions before beginning the upgrade of your development environment:

- 1 Perform a partial synchronization for mobile users, sending all transactions to the server database.

CAUTION: Mobile users must make no further changes to their local databases until the upgrade has been completed. Any changes made before the upgrade has been completed will be lost when they are reinitialized following the upgrade.

- 2 Verify that mobile clients have synchronized and that all changes have been merged into the server database:
 - a Check that no transaction files remain in the synchronization inbox and outbox for any mobile user. The synchronization inbox for each user is on the Siebel Server.

Microsoft Windows inboxes will be in the
`C:\SIEBSRVR_ROOT\docking\mobileusername\` directory.

Transaction files are in the format `number.dx`; for example, `00000023.dx`.
 - b Log onto a Siebel eBusiness Application, such as Call Center, as the Siebel Administrator. Use the Server Administration - Server Tasks screen to make sure that each Transaction Merger task has successfully completed.
 - c Verify that Workflow Monitor and Workflow Action agents have processed all pending requests. If Workflow Manager has completed successfully, the `S_ESCL_REQ` table should not have any rows.
- 3 To prevent synchronization of mobile clients with the database server, stop or disable all Siebel Remote components on all Siebel Servers, as described in *Siebel Remote Administration Guide, MidMarket Edition* and in *Siebel Server Administration Guide, MidMarket Edition*.
- 4 Disconnect all Web clients from the database server by stopping the appropriate Application Object Managers, as described in *Siebel Server Administration Guide, MidMarket Edition*.

- 5 Make sure that Dedicated Web clients have disconnected from the database server.

The method you use to do this will depend on your database. For example, with an Oracle RDBMS, you would stop the primary listener. However, all RDBMS types require starting the database in restricted mode. Refer to the documentation that you received from your RDBMS vendor for more information.

Upgrading Your RDBMS Software

The new release of Siebel eBusiness Applications might require you to upgrade your RDBMS server software. Review the information in *Siebel System Requirements and Supported Platforms* to determine if a database server upgrade is required. Refer to the documentation provided by your database vendor for specific instructions on performing the database upgrade.

CAUTION: If your system does not meet the requirements specified in *Siebel System Requirements and Supported Platforms*, your upgrade will fail.

- If you previously installed DB2 UDB, you must upgrade to the latest DB2 UDB version before upgrading to Release 7.x. Verify that you have updated the database to the current fixpack level as described in *Siebel System Requirements and Supported Platforms*.
- If you are performing an upgrade for DB2 UDB, you must use a DB2 database with 4-KB, 16-KB and 32-KB table spaces defined on it. Otherwise, your upgrade will not complete successfully.
- The `NextSeq` user defined function (UDF) from Release 6.x was replaced by an alternate version for 7.x, so your DBA needs to verify that only the current version of the UDF is installed. The new function definition accepts only one parameter, and the old one is dropped.

Carefully read the relevant chapters of *Siebel Server Installation Guide for Microsoft Windows, MidMarket Edition* for the RDBMS software you plan to configure and install, and for instructions for installing `NextSeq`.

Make sure that the RDBMS upgrade was successful before proceeding with the upgrade of your Siebel database.

NOTE: Be sure to upgrade your client database connectivity software on all upgraded machines to the version specified in *Siebel System Requirements and Supported Platforms*.

Verifying Database Server Configuration

Before you begin your upgrade, you must verify your database configuration. Later steps of the upgrade process are database-intensive and demand that your database have sufficient resources available. While specific requirements vary by database platform, the consequences of exceeding available resources are the same across platforms: a halted upgrade that requires you to allocate time to adjust the environment and then resume the upgrade.

NOTE: If you want to change your database platform in conjunction with your upgrade, please contact Siebel Technical Services for assistance.

When you upgrade from earlier versions of Siebel eBusiness Applications to Release 7.5, expect database growth in the range of 30%-60% across all platforms. The amount by which it grows depends on a number of factors, such as the size of your database prior to upgrade and the RDBMS software you use.

For estimates of growth percentages for specific RDBMS platforms and releases, see [Table 8 on page 69](#).

For all platforms, the growth percentage will increase depending on how you size your database and configure default storage for database table spaces.

For example, if you set the default storage for your initial or next extent in a given Oracle table space to 10 KB, that table space will grow by a smaller percentage than if you set it to 100 KB.

Table 8. Siebel Database Expected Growth During Upgrade

Release From Which You Are Upgrading	DB2 UDB	MS SQL Server	Oracle
Release 6.3x	4-K table space: 200 % 16-K table space: 400 % 32-K table space: reduced by 50 %	N/A	tables 40 %-60 % indexes 70 %-80 %
Release 6.2x	4-K table space: 200 % 16-K table space: 400 % 32-K table space: reduced by 50 %	data 40 %-50 % index 60 %-80 %	tables 40 %-60 % indexes 70 %-80 %
Release 6.0x	4-K table space: 200 % 16-K table space: 400 % 32-K table space: reduced by 50 %	data 40 %-50 % index 60 %-80 %	tables 40 %-60 % indexes 70 %-80 %
Release 5.6x	4-KB table space: 40 % Long table space: 250 % 8-KB table space to 16-KB table space: 350 %	N/A	40 %-80 %
Release 5.5x	N/A	data 50 %-60 % index 80 %-100 %	40 %-80 %
Release 5.0x	Same as from version 5.6x	data 50 %-60 % index 80 %-100 %	40 %-80 %

NOTE: Actual expected growth may vary widely from these estimates, depending on your database configuration, row size of tables, and data content.

Review the configuration requirements for your specific database platform and make sure that your development database server configuration meets or exceeds them.

- For server configuration requirements for IBM DB2 UDB, see [“DB2 Universal Database Server Configuration” on page 76](#).
- For server configuration requirements for Microsoft SQL Server, see [“Microsoft SQL Server Database Server Configuration” on page 70](#).
- For server configuration requirements for Oracle, see [“Oracle Database Server Configuration” on page 73](#).

Microsoft SQL Server Database Server Configuration

This section contains guidelines for obtaining optimal performance from the Microsoft SQL Server database for use with Siebel eBusiness Applications.

NOTE: These settings should be used only as guidelines for your upgrade configuration. Your final settings will vary based on the server hardware configuration, the number of users, and the type of workload.

Additional information on the configuration of Microsoft SQL Server is available in the Microsoft documentation, information provided by your hardware vendor, and other sources. For additional information concerning tuning options for Microsoft SQL Server, refer to the Microsoft documentation.

CAUTION: Never make changes to your Siebel database schema that are not documented in this guide or elsewhere in the *Siebel Bookshelf*. Otherwise, you may corrupt your entire system and thereby render it unsupportable.

If you are upgrading from Siebel eBusiness Applications Release 5.x or 6.x, you must upgrade the Microsoft SQL Server database from 7.x to SQL Server 2000, using Microsoft's documentation and tools. After you have upgraded SQL Server, be sure that you configure it according to the following recommendations before proceeding with the upgrade.

NOTE: The development database must use binary sort order. Production databases are not constrained by this requirement. This is not the default for Microsoft SQL.

Temporary Database Space

This is the database that Microsoft SQL Server uses for temporary space needed during execution of various queries.

- The default size of `TEMPDB` is too small for almost all production installations. Make `TEMPDB` as big as the biggest table in the Siebel database, or half the size of the Siebel database.
- Make sure that the files used by `TEMPDB` are configured to allow auto-growth. This allows SQL Server to expand the temporary database as needed to accommodate your activity. Alternatively, you can set `MAXSIZE` to the size of the biggest table or to 50% of the size of the Siebel database.
- Put `TEMPDB` on a separate drive for performance reasons.

NOTE: Execute `dbcc shrinkdatabase` against `TEMPDB` before starting the upgrade.

Configuration Parameters

Table 9 describes Microsoft SQL Server database configuration parameters that must be set for the upgrade. Before upgrading a MS SQL Server database, make sure that your development database server meets these parameters. For parameters not listed in this table, Siebel Systems recommends that you accept the default settings.

Table 9. Microsoft SQL Configuration Parameters

Parameter	Setting/Comment
Max. degree of parallelism	0
Cost threshold for parallelism	5
Fill factor (%)	90
Index create memory (KB)	0

NOTE: Most of these parameters do not differ from the default settings.

Siebel Database Options

Set the following Siebel database options to ON (enabled) for the upgrade. After your development environment upgrade is complete, you will need to reset these options to their installation settings, as described in Chapter 5, “Development Environment Postupgrade Tasks.”

truncate log on chkpt. This option should be set to ON (enabled) for upgrade only. Also, for upgrade only, execute the alter database command against Siebel database specifying set recovery simple.

NOTE: You must revert to the original database recovery setting after the upgrade.

torn page detection. Set this option to ON (enabled).

auto create statistics. Set this option to ON (enabled).

auto update statistics. Set this option to ON (enabled).

Rebuild Clustered Indexes

If you have large tables that you use extensively (such as `S_EVT_ACT`, `S_CONTACT`, `S_OPTY`, `S_OPTY_POSTN`, `S_ORG_EXT`), use the MS SQL Server `create index` command with `drop_existing` clause to rebuild large tables with high fillfactor (60%-70%).

Update Statistics

The cost-based optimizer in Microsoft SQL Server uses statistics about tables and indexes to compute the most efficient access plans. When statistics become inaccurate, which can happen for tables with high insertion rates, high deletion rates, or both, and for associated indexes, the performance of database operations can degrade dramatically.

To update statistics, run `sp_updatestats` to refresh statistical information in the Siebel database.

Oracle Database Server Configuration

Before upgrading an Oracle database, verify the following:

- Upgrade your Oracle database and client software to the version referenced in *Siebel System Requirements and Supported Platforms*, using Oracle's tools and documented procedures.
- Make sure that the system table space has sufficient room for growth, since a number of additional objects are created during the upgrade.

Table Spaces

Allow for growth of the database during the upgrade. Make sure that table spaces have enough free space to accommodate expected growth as estimated in [Table 8 on page 69](#).

pctincrease. For upgrades, you should have a high value `pctincrease` for the table spaces that contain application table spaces and indexes.

pctfree. Before you start the upgrade, rebuild some of your larger tables with a large value of `pctfree` for the larger tables (30 or higher). Examples of large tables are:

- `s_contact`
- `s_evt_act`
- `s_srv_req`
- `s_src`
- `s_org_ext`
- `s_addr_org`

NOTE: The reason that you need to increase `pctfree` before the upgrade is that many new columns are added to these tables in Release 7.x. Migrating data into the new columns during the upgrade is likely to cause row chaining, which will cause performance degradation.

Database Parameters

If the database server is a multiple-CPU machine, you can take advantage of parallel index creation. The parameter in the master file, Oracle Parallel Index, defaults to the value `N` for development environment upgrades. If your database server is a single-CPU machine, the value of this parameter must be set to `N`, or the operation will fail. However, if your database server has more than one CPU, you can change this parameter to `Y` to yield significance performance benefits.

If parallel index creation is supported at the database server level, you can modify the master configuration file found in `SIEBSRVR/bin` directory. The master file that you need to modify depends on the Siebel version that you are upgrading from, and whether you are upgrading the Siebel Repository or your custom repository (for example, `master_upgrep_dev_601.ucf`).

The DBA should set appropriate values for the following `init.ora` parameters, depending on the number of CPUs on the database server:

- `parallel_max_servers`

NOTE: The parameter `parallel_max_servers` must be set greater than 1 to enable parallelism.

- `parallel_min_servers`
- `db_block_lru_latches`

The other `init.ora` parameters that the DBA should carefully choose are:

- `sort_area_size`
- `sort_area_retained_size`
- `log_buffers`

For more information, see your Oracle documentation.

Rollback Segments

Rollback segments should be appropriately sized so that the largest of transactions can be accommodated. The upgrade may affect some of the largest tables in your implementation of Release 7.x, causing them to grow by as much as 40%.

The shared pool size should be sufficiently large.

DB2 Universal Database Server Configuration

This section provides guidelines for obtaining optimum performance from a DB2 Universal Database. These guidelines will be useful to a broad segment of customers.

Before upgrading a DB2 database, verify that your development database server meets or exceeds the following OLTP parameters:

- You have at least 25 primary and 100 secondary logs of at least 32-MB log size.
- Locklist parameters are set to at least 5,000, and preferably 15,000.
- Maxlocks should be set to 20.
- DMS table space has at least 25% of free pages. If it does not, you will need to increase table space size by adding containers.
- The file system has sufficient space to allow your DMS table space to grow.

In addition, verify that the table spaces are not near their capacity. This can be done by connecting to the database and issuing the following command:

```
db2 list tablespaces show detail
```

Upgrade Instance

After upgrading your RDBMS software (for DB2 UDB, follow the instructions provided by IBM), upgrade your DB2 UDB instance.

To upgrade the instance

- Run the following command on the database server to upgrade to the current version of your RDBMS software:

```
db2updv7 -d DB_NAME
```

Verifying DB2 on AIX Permissions

Before executing the Siebel Database Upgrade if you are running DB2 UDB on the AIX platform, perform the following steps:

- 1 Navigate to the instance home directory.
- 2 Use the following command to verify that the directory `sqlllib/function/routine/sqlproc` has write permission for the group:

```
ls -ld sqlllib/function/routine/sqlproc
```

- 3 To authorize group write permission, enter the following command:

```
chmod g+w sqlllib/function/routine/sqlproc
```

Increasing the Number of ODBC Statement Handles

DB2 UDB can quickly run out of ODBC statement handles, depending on the number of business objects your enterprise uses. Because it is difficult to know how many business objects your users actually use, you should increase this number automatically each time you upgrade the DB2 UDB client, or when rebinding database utilities.

Siebel Systems recommends that you increase the number of CLI packages to six by rebinding the CLI packages, using the special DB2 `CLIPKG` bind option.

To rebind the CLI packages

- 1 Navigate to `C:\sqlllib\bnd` from a DB2 Command window.
- 2 Connect to the DB2 UDB database.
- 3 Enter the following command:

```
db2 bind @db2cli.lst blocking all grant public clipkg 6
```

For more information about the DB2 bind command and the `CLIPKG` option, refer to IBM DB2 documentation.

Upgrade-Specific Parameters

The default settings of the parameters in this section should be adjusted for upgrading to Release 7.5. The values recommended in the following pages are guidelines only, and your environment may require adjustments to these values.

After your upgrade has been completed, and prior to running Release 7.5 in a production environment, you must adjust the DB2 parameters described in this section to those values recommended in *Siebel Server Installation Guide for Microsoft Windows, MidMarket Edition*. Otherwise, your Siebel Database Server may not provide optimal performance.

See your IBM DB2 technical documentation for additional information on DB2 parameters.

DB2 Database Manager Configuration Parameters

You can set the database configuration parameters using the `update database manager configuration` command of the DB2 Command Line Processor or using the DB2 Control Center.

NOTE: Refer to your IBM DB2 UDB documentation for more information on modifying the database configuration parameters.

Table 10 provides guidelines for setting DB2 Database Manager configuration parameters specifically for an optimal upgrade of your Siebel Database. Set these parameters for each DB2 instance. Use the configuration information below for the listed parameters. For parameters not listed in this table, accept the default settings.

Table 10. DB2 Database Manager Configuration Parameters

Parameter	Explanation	Setting
SHEAPTHRES	Sort heap threshold (4 KB)	Double the value allocated for SORTHEAP. See Table 12 on page 80.
ASLHEAPSZ	Application support layer heap size (4 KB)	15
MON_HEAP_SZ	Database monitor heap size (4 KB)	128 (minimum)

Table 10. DB2 Database Manager Configuration Parameters

Parameter	Explanation	Setting
UDF_MEM_SZ	UDF shared memory set size (4 KB)	256
RQRIOBLK	Max. requester I/O block size (bytes)	65535
QUERY_HEAP_SZ	Query heap size (4 KB)	16384
KEEPDARI	Keep DARI process	YES
QUERY_HEAP_SZ	Query heap size (4 KB)	16384
MAX_COORDAGENTS	Maximum number of coordinating agents	MAXAGENTS

db2set Parameters

Use the `db2set` command to set the parameters (for example, `db2set DB2_RR_TO_RS = YES`) referenced in [Table 11](#). (Under Windows, you would access this through the DB2 Command Window, accessible from the DB2 for Windows 2000 client.)

Table 11. db2set Parameters

Parameter	Explanation	Setting
DB2_RR_TO_RS	Improves DB2 performance with Siebel eBusiness Applications. <i>Set to YES only in production environment servers.</i>	YES
DB2_MMAP_WRITE	Recommended setting only; you should evaluate this setting for your particular configuration and environment.	OFF
DB2_MMAP_READ	Recommended setting only; you should evaluate this setting for your particular configuration and environment.	OFF
DB2_CORRELATED_PREDICATES	When set to ON, the optimizer is able to determine whether predicates in a query are related, which permits DB2 to calculate the filter factor more accurately.	ON
DB2_INDEX_2BYTEVARLEN	This parameter must always be set to ON. Otherwise, you will not be able to create indexes with columns greater than 255 bytes.	ON

Table 11. db2set Parameters

Parameter	Explanation	Setting
DB2_PIPELINED_PLANS	Tells the DB2 optimizer to favor pipeline execution plans; in other words, plans which are left deep and have no temporary result sets.	ON
DB2_INTERESTING_KEYS	Limits the number of execution plans generated by the DB2 optimizer.	ON
DB2_PARALLEL_IO	Useful when using RAID devices. For more information, refer to relevant IBM documentation.	ON
DB2_STRIPED_CONTAINERS	Useful when using RAID devices. For more information, refer to relevant IBM documentation.	ON

NOTE: After changing any of these settings, perform a `db2stop/db2start` to implement the changes in your DB2 database.

DB2 Database Configuration Parameters

The database configuration parameters can be set using the `update database configuration` command of the DB2 Command Line Processor or using the DB2 Control Center.

NOTE: See the IBM DB2 technical documentation for more information on modifying the database configuration parameters.

Set the parameters in [Table 12](#) for *each* database within an instance on which you run your Siebel eBusiness Applications. For other parameters of the same type, accept the default settings.

Table 12. DB2 Database Configuration Parameters

Parameter	Explanation	Setting
DFT_DEGREE	Degree of parallelism (1 = turn query parallelism off)	1
DFT_QUERYOPT	Default query optimization class	3
DBHEAP	Database heap (4 KB)	7429

Table 12. DB2 Database Configuration Parameters

Parameter	Explanation	Setting
CATALOGCACHE_SZ	Catalog cache size (4 KB)	5558
LOGBUFSZ	Log buffer size (4 KB)	512 (For Windows, set this to 256.)
UTIL_HEAP_SZ	Utilities heap size (4 KB)	5000
LOCKLIST	Maximum storage for lock list (4 KB)	5000 (The setting should never be smaller than this, but may be increased.)
APP_CTL_HEAP_SZ	Maximum applications control heap size (4 KB)	152 (Recommended size may increase or decrease with the number of users.)
STMTHEAP	SQL statement heap (4 KB)	8192
SORTHEAP	Sort list heap (4 KB)	20,000 – 40,000 Recommended size; this may increase or decrease depending on the amount of memory in the database server machine and the size of the data. A 20,000 setting allows SORTHEAP to increase up to 80 MB.
APPLHEAPSZ	Default application heap (4 KB)	2500 (Recommended size may increase or decrease with the number of users.)
STAT_HEAP_SZ	Statistics heap size (4 KB)	8000
MAXLOCKS	Percentage of lock lists per application	5
LOCKTIMEOUT	Lock timeout (sec.)	90 to 150
CHNGPGS_THRESH	Changed pages threshold	5
NUM_IOCLEANERS	Number of asynchronous page cleaners	Number of CPUs
INDEXSORT	Index sort flag	YES
SEQDETECT	Sequential detect flag	YES

Table 12. DB2 Database Configuration Parameters

Parameter	Explanation	Setting
LOGRETAIN	Sequential or circular log files	NO Setting this parameter to YES means that log files will be archived and the potential exists for the file system contained the log files to fill up if you do not move or archive the logs.
AVG_APPLS	Average number of active applications	1
MAXFILOP	Maximum DB files open per application	500
LOGFILSIZ	Log file size (4 KB)	8000-16000
LOGPRIMARY	Number of primary log files	25-50 The value of LOGPRIMARY and LOGSECOND together may not exceed 128.
LOGSECOND	Number of secondary log files	Accept the DB2 UDB default value; increase this value if secondary log files are required for your deployment.
SOFTMAX	Triggers buffer pool flushing	50
NUM_IOSERVERS	Number of disks on which the database resides	Number of disks

Installing the Stored Procedures and User-Defined Functions

NOTE: If you are performing an upgrade from Release 7.0.x to Release 7.5, skip this procedure. You already completed this step during your upgrade to Release 7.0.x.

If you are upgrading from Release 5.x or 6.x and your RDBMS is DB2 UDB, you must install the stored procedures and user-defined functions (UDFs) on the database server. To do this, you must first transfer them to the database server, and have installed the database server components. (For information on installing database server components, refer to the chapter on installing the Siebel Database Server for DB2 Universal Database in *Siebel Server Installation Guide for Microsoft Windows, MidMarket Edition*.)

The user-defined functions (UDFs) and stored procedures must be transferred to and installed on the database server to support the Siebel product. Any method that transfers the necessary files to the correct location on the database server is acceptable.

NOTE: Compiling stored procedures in DB2 creates .dll files. You must copy these files to the database server machine's `sqllib\function` directory.

To copy and install the stored procedure code, follow the procedure appropriate to your database server platform. You first must delete the old stored procedures. Then you need to install the Siebel stored procedure library on the DB2 database server host.

Before you perform this procedure, make sure that you have followed IBM documentation regarding upgrading your DB2 database software to version 7.x. As part of your DB2 upgrade, you must apply the DB2 fix pack that comes with Siebel software. For information on how to perform basic DB2 tasks, refer to the relevant IBM documentation.

To delete the old stored procedures

- Delete the old stored procedures from the `sqllib\function` directory:
 - On the Windows Database Server:

<code>nextseq.dll</code>	<code>mapping.dll</code>
<code>nextseqd.exe</code>	<code>updt2col.dll</code>
<code>siebstat.dll</code>	<code>updt_addr_org_sp.dll</code>
<code>siebsdmy.dll</code>	<code>updt_addrorg_intsp.dll</code>
<code>siebtrun.dll</code>	<code>updt_addr_per_sp.dll</code>
<code>siebfct.dll</code>	

- On the UNIX Database Server:

<code>nextseq</code>	<code>updt2col</code>
<code>nextseqd</code>	<code>siebtrun</code>
<code>siebstat</code>	<code>siebfct</code>
<code>siebupgl</code>	<code>updt_addr_org_sp</code>
<code>siebsdmy</code>	<code>updt_addr_per_sp</code>
<code>mapping</code>	<code>updt_addrorg_intsp</code>

To copy and install the stored procedure code

- 1** Install the Siebel stored procedure library on the DB2 database server host.
- 2** Log onto the Siebel source installation machine, and navigate to the source installation subdirectory that contains the Siebel Database installation objects.

The directory that contains the file to install (`siebproc`) is

```
DBSRVR_ROOT \DB2UDB\SIEBPROC\DBSRVR_OS
```

where:

DBSRVR_OS = the operating system your database server runs on; for example, WINNT or aix.

- 3** Put the `siebproc` file (on Windows, this is called `siebproc.dll`) into the `FUNCTION` subdirectory within the DB2 UDB instance directory (where DB2 UDB is installed) on the Siebel Database Server.

For example, on Windows, this location might be `C:\SQLLIB\FUNCTION`.

Preparing the Database for a DB2 Upgrade

You need to perform the following procedures to prepare your database for a DB2 UDB upgrade:

If you are upgrading from Release...	Perform the following procedure...
5.x, 6.x, or 7.0.x	Create DB2 temporary table spaces and buffer pools.
5.x, 6.x, or 7.0.x	Increase 4-KB table spaces.
5.x	Create 16-KB table spaces and buffer pools.
6.x or 7.0.x	Increase 16-KB table spaces.
5.x or 6.x	Create 32-KB table spaces and buffer pools.
7.0.x	Increase 32-KB table spaces and buffer pools.
5.x, 6.x, or 7.0.x	Determine table space page size requirements.
5.x, 6.x, or 7.0.x	Verify that the Application Development Client/Tool is installed on your database server.

Creating DB2 Temporary Table Spaces and Buffer Pools

If your RDBMS is DB2 UDB, verify that you have 16-KB and 32-KB temporary table spaces to use for sorting and other SQL processing. Both the 16-KB and 32-KB temporary table spaces require dedicated buffer pools.

To create a 16-KB temporary table space

- 1 Create a 16-KB buffer pool with at least 5,000 16-KB pages.
- 2 Create a 16-KB temporary table space as system managed space (SMS) that can be expanded to 2 GB of storage.

To create a 32-KB temporary table space

- 1 Create a 32-KB buffer pool with at least 1,000 32-KB pages.
- 2 Create a 32-KB temporary table space as SMS that can be expanded to 2 GB of storage.

Increasing DB2 4-KB Table Space

If you are upgrading from Release 5.x, 6.x, or 7.0.x and your RDBMS is DB2 UDB, you must increase the size of your 4-KB table space.

To increase the size of your 4-KB table space

- Increase the 4-KB table space using the DB2 tools of your choice. For sizing requirements, see [Table 8 on page 69](#).

This completes the table space allocation for your 4-KB table space.

Creating DB2 16-KB Table Space and Buffer Pool

If you are upgrading from Release 5.x and your RDBMS is DB2 UDB, create a new 16-KB table space and buffer pool, to which the data in your previous installation's 8-KB table space will be migrated.

NOTE: Skip this step if you are upgrading from Release 6.x or 7.0.x, because you already created your 16-KB table space and buffer pool. Instead, perform the task described in [“Increasing DB2 16-KB Table Space” on page 87](#).

To create 16-KB table space and buffer pool

- 1 Create a 16-KB buffer pool, with a number of pages equal to at least twice the number of pages in the 8-KB buffer pool in your previous installation.

NOTE: If your system has limited RAM, you may want to reduce the size of your current 8-KB buffer pool or increase the size of your swap file before upgrading; this will prevent an upgrade failure due to lack of memory.

- 2 Create a 16-KB table space using the DB2 tools of your choice.

This completes creation of your new 16-KB table space and buffer pool.

Increasing DB2 16-KB Table Space

If you are upgrading from Release 6.x or 7.0.x and your RDBMS is DB2 UDB, you must increase the size of your 16-KB table space.

NOTE: Skip this step if you are upgrading from Release 5.x.

To increase the size of your 16-KB table space

- Increase the 16-KB table space using the DB2 tools of your choice. For sizing requirements, see [Table 8 on page 69](#).

This completes the table space allocation for your 16-KB table space.

Creating DB2 32-KB Table Space and Buffer Pool

If you are upgrading from Release 5.x or 6.x and your RDBMS is DB2 UDB, you must create a new 32-KB table space and buffer pool in order for the upgrade to complete successfully.

NOTE: Skip this step if you are upgrading from Release 7.0.x (for example, Release 7.0.3 or 7.0.4), because you already created your 32-KB table space and buffer pool. Instead, perform the task described in [“Increasing DB2 32-KB Table Space” on page 88](#).

To create 32-KB table space and buffer pool

- 1** Create a 32-KB buffer pool, with the number of pages equal to at least 1,000 32-KB pages.
- 2** Create a 32-KB table space using the DB2 tools of your choice.

This completes creation of your new 32-KB table space and buffer pool.

Increasing DB2 32-KB Table Space

If you are upgrading from Release 7.0.x and your RDBMS is DB2 UDB, you must increase the size of your 32-KB table space.

NOTE: Skip this step if you are upgrading from Release 5.x or 6.x.

To increase the size of your 32-KB table space

- Increase the 32-KB table space using the DB2 tools of your choice. For sizing requirements, see [Table 8 on page 69](#).

This completes the table space allocation for your 32-KB table space.

Determining Table Space Page Size Requirements for DB2 UDB

There are four standard database managed table spaces (DMS) that hold Siebel tables and indexes—a 4-KB, 16-KB, 32-KB table space, for various sized tables, and a table space to hold indexes. However, in some cases, you may have Siebel tables in custom table spaces.

If a custom table has an estimated page size greater than its current table space page size, it will not fit in its table space after the upgrade, and the upgrade will fail. The following utility will determine whether each of your current custom tables will increase in size to the point that it will require a move to a larger table space.

This utility must be run prior to the upgrade, before any steps of the upgrade are executed.

To determine table space page size requirements for DB2 UDB

- 1 From the `siebsrvr/bin` directory, type the following command line:

```
tblsize /U $Tableowner /P $Password /C $ODBCDataSource /F
$DDLFilename /B $DefaultTablespace /X $DefaultIndexspace /K
$16Ktablespace /V $32Ktablespace /Q $ReportFilename /L
$LogFilename
```

where:

- `Tableowner` = Tableowner
- `Password` = Tableowner password

- `ODBCDataSource` = Data source of the database
 - `DDLFilename` = Name of the DDL file (This file is called `ddl.ct1`, and it is located in the `dbsrvr/db2udb` directory.)
 - `DefaultTablespace` = Name of the 4-KB page standard Siebel table space
 - `DefaultIndexspace` = Name of the standard Siebel index space
 - `16Ktablespace` = Name of the 16-KB page standard Siebel table space
 - `32Ktablespace` = Name of the 32-KB page standard Siebel table space
 - `ReportFilename` = Name of the report generated by the utility
 - `LogFilename` = Name of the log file (The default name is `custtbl1.log`.)
- 2** Review the report generated by the utility to determine if the status of the estimated table pagesize postupgrade is larger than the size of the custom table pagesize.

An example of the report generated by this utility is provided below:

```
Table Name = S_EVT_ACT
Custom Tablespace Id = 5
Custom Tablespace Name = CUST_TBS_EVT_ACT
Custom Tablespace Pagesize = 4096
Estimated Table Pagesize (postupgrade) = 5067
Status = Does not fit in its custom tablespace
```

- 3** For each table which has Status: Does not fit in its custom tablespace, you must create a larger custom table space which is larger than the estimated table pagesize postupgrade.

- 4 Move the tables from their old table spaces to their new table spaces by running `ddlmove`.

DDLMOVE is a utility for moving tables from one table space to another table space. This utility is located under the `siebsrvr/BIN` directory.

To run `ddlmove`, submit the following arguments:

```
ddlmove /U $Tableowner /P $TablePassword /C $ODBCDataSource /E  
$Stop_on_DDL_Error /G $Grantee /B $Tablespace /X $IndexTablespace  
/M $TableName /L $LogFilename /Z $UCS2Database
```

where:

- `Tableowner` = Table owner of the database (Required)
- `TablePassword` = Password of the table owner of the database (Required)
- `ODBCDataSource` = Data source of the database (Default environment variable: `SIEBEL_DATA_SOURCE`)
- `Stop_on_DDL_Error` = Stop on DDL Error (Default: Y)
- `Grantee` = Grantee for tables
- `Tablespace` = Name of the table space that you are moving the table to
- `IndexTablespace` = Name of the index space that you are moving the table to
- `TableName` = Table Name Like Support (Default: N)
- `LogFilename` = Name of the log file (The default name is `ddlmove.log`.)
- `UCS2Database` = (Default: N)

NOTE: If there are problems reported by the sizing utility, you must resolve the table space page sizes before you proceed with the upgrade.

Verifying Installation of the DB2 UDB Application Development Client

If you are upgrading to Release 7.5 from Release 5.x, 6.x, or 7.0.x and your RDBMS is DB2 UDB, you must verify that the DB2 UDB Application Development Client is installed on your Siebel Database Server before proceeding with the upgrade. To do this, navigate to the appropriate directory on the Siebel Database Server and verify that the DB2 UDB Application Development Client is installed.

[Table 13](#) lists the DB2 UDB Application Development Client components that must be installed on your Siebel Database Server depending on your platform. Use this table to verify that you have the correct components installed.

Table 13. DB2 UDB Application Development Client Components

Platform	DB2 UDB Application Development Client Components
Windows	DB2 Application Development Client
AIX	db2_07_01.adt.rte 7.1.0.xx COMMITTED Application Development Tools db2_07_01.adt.samples 7.1.0.xx COMMITTED ADT Sample Programs
HP	DB2V7SKL 7.1.0.40 Application Development Tools for HP-UX
Solaris	application db2adt71 Application Development Tools (ADT) (PTF 1720500-00) application db2adts71 ADT Sample Programs (PTF 1720500-001)

If the DB2 Application Development Client is not installed, you must install it. For more information, refer to the relevant IBM documentation.

Preparing Tables for Upgrade

Take the following measures to prepare your tables for the upgrade.

Preserving Custom Indexes on Tables

Release 7.x upgrade drops and recreates custom indexes on base tables. However, custom indexes on interface tables are not recreated during the upgrade, but they can be restored when the upgrade is complete.

NOTE: Custom indexes may need to be changed to reflect schema changes. You should reevaluate custom indexes for applicability in the new release.

For more information about applying custom indexes, see *Siebel Tools Reference, MidMarket Edition*.

CAUTION: If your DBA created custom indexes in your current schema that were not defined through Siebel Tools in the Siebel repository, these indexes will be dropped during the upgrade process.

Considerations for Clustered Indexes

If you created clustered indexes on base tables, and Release 7.x introduces a different clustered index on the same table, the upgrade process will recreate custom indexes as nonclustered and create the Siebel index as clustered.

For IBM DB2 UDB, indexes that reside on tables that have been defined with append mode enabled will be recreated as nonclustered indexes during the upgrade. Tables that have been created with append mode enabled cannot contain [support] clustered indexes.

Disabling Customized Triggers

Release 7.x does not support customized triggers. If you have created customized triggers on your Siebel base tables, please disable them before you perform the upgrade. You will then need to recreate them after the upgrade is finished.

Dropping Customized Views

If you have created customized views on your Siebel base tables, you must drop them before you perform the upgrade. If they are still applicable after the upgrade, you will need to recreate them after the upgrade is finished.

Identifying DB2 UDB Long Columns for Truncation

In Release 7.5, the maximum length for DB2 UDB long columns with a type of varchar has reduced to 16350 from 16383. Long columns of type varchar that exceed 16,350 will be truncated.

To prevent a data truncation error that may cause transaction processing (`txnproc`) or transaction routing (`txnroute`) to fail, perform the following steps to identify these columns and reduce the data in these columns.

To identify and reduce the length of long varchar columns

- 1 From any shell, open the script `chk16350.bat`, and edit the following parameters as appropriate for your deployment:

`SRC_USR` = username of the source database

`SRC_PSWD` = password for the source database

`SRC_TBLO` = table owner of the source database

`SRC_TBLO_PSWD` = table owner password for the source database

`SRC_ODBC` = ODBC data source name of the source database (edit the value “CHANGE_ME”)

`SRC_REPOSITORY_NAME` = repository name of the source database

`DBSRVR_ROOT` = directory where you installed the dbsrvr component of the Siebel Server (edit the value “CHANGE_ME”)

`SIEBEL_ROOT` = directory where you installed the siebsrvr component of the Siebel Server (edit the value “CHANGE_ME”)

`VALID_RESULTS_DIR` = directory where you want the output files to be generated (edit the value “CHANGE_ME”); this must be an existing directory

This script produces two files:

- **long_trunc_cols.rpt.** This report identifies all long varchar columns that are longer than 16350 characters.
 - **update_trunc.sql.** This SQL file will generate update statements that truncate identified columns to 16350 characters.
- 2 Reduce the data in these columns using either of the following methods:
 - Manually review the columns in the `long_trunc_cols.rpt` report and manually reduce the size of each column identified.
 - Run `update_trunc.sql` using the DB2 command line processor.

CAUTION: If you do not truncate or otherwise reduce the data in these columns, you will receive a “data truncated” error, and transaction processing and transaction routing may fail.

Preparing the Siebel Repositories for Upgrade

Four separate repositories are used during the development upgrade process:

- **Your existing development repository.** To prevent a naming conflict, before you run the upgrade, rename your existing development repository (*Siebel Repository*) to *Prior Customer Repository*. After the upgrade, your new Release 7.5 development repository will be given the name *Siebel Repository*.
- **Three new repositories.** These are automatically loaded when you run the upgrade wizard:
 - Prior standard repository (Release 5.x or 6.x)
 - New standard repository (Release 7.5)
 - New customer repository (Release 7.5) which, after the merge, will become the customized 7.5 repository.

Before these repositories are loaded, verify that your existing repositories do not use the names reserved for the upgrade process:

- **New Customer Repository.** Your new custom repository, which results from the merge, and includes your custom configurations.
- **New Siebel Repository.** The new Siebel standard repository.
- **Prior v5.x Siebel Repository** (for upgrades from Release 5.x).
- **Prior v6.x Siebel Repository** (for upgrades from Release 6.x).
- **Prior v7.x Siebel Repository** (for upgrades from Release 7.x).

Renaming Your Existing Development Repository

Use the version of Siebel Tools that matches your old version of Siebel eBusiness Applications (for example, Siebel Tools Release 6.x for upgrades from Release 6.x). Connect directly to the development database server, and then perform the following steps to make sure that your existing repositories follow the correct naming conventions.

To rename the repository

- 1** Using the appropriate prior version of Siebel Tools, connect to the Siebel Database Server.
- 2** Change the name of the existing repository, as described below:
 - a** In the Object Explorer, click the Types tab.
 - b** Click Repository.
 - c** In the Repository view, click Name.
 - d** Locate the appropriate repository in the list applet and rename it `Prior Customer Repository`.
- 3** Step off the list to commit the record to the database.

NOTE: The upgrade process will verify the repository names. If no repository is named `Prior Customer Repository`, the Siebel Repository will be renamed to `Prior Customer Repository` in the target database so that the upgrade will execute properly.

For more information about renaming repositories, refer to *Siebel Tools Reference, MidMarket Edition*.

Recording Dock Objects and Visibility Rules

Modified visibility rules will be dropped during the upgrade. Manually record your changes to dock object visibility rules so you can evaluate the need to reapply the changes after the upgrade is complete.

Dock objects and visibility rules created by using Docking Wizard will be preserved unless they become invalid after the upgrade. Manually record any changes that you made through the Docking Wizard so that you can evaluate the need to reapply the changes after the upgrade is complete.

NOTE: Changes to visibility rules require the assistance of Expert Services.

Increasing Database File Size

If your RDBMS is MS SQL Server, you should increase your database file size by resetting the `Autogrowth` parameter to between 25% and 50%. Failure to do this could diminish upgrade performance and possibly impact the success of your upgrade.

NOTE: Failure to increase your database file size might result in an entry in the `ddctl` log files.

Preparing Application Data for Upgrade

If you use one of the following Siebel eBusiness Applications, you need to perform preupgrade procedures to prepare your data for upgrade to Release 7.x:

- Siebel eChannel
- Quotes
- Timesheet
- Calendar

Siebel eChannel

NOTE: If you are upgrading from Release 7.0.x to Release 7.5, you do not need to perform this procedure.

Before upgrading to Release 7.x, Siebel eChannel customers who have modeled their partners as both Accounts and Divisions (or Organizations) should merge these two records to make sure that only one record will exist for each partner company in the new single party model. Similarly, partner employees that have been modeled as Contacts and Employees should be merged to result in a one-person record in the upgraded database. There is a tool that will help customers find and populate a temporary table with the matching partner company and employee records. Records that have been identified as matches will be merged during the upgrade, creating a single record for each partner company or partner employee. For information about this tool, contact Siebel Professional Services.

Quotes

NOTE: If you are upgrading from Release 7.0.x to Release 7.5, you do not need to perform this procedure.

To make sure quotes that use products as solutions in Release 6.x will be upgraded to Release 7.x, verify that the Model Product field is unchecked.

In Siebel Quotes 6.x, the Discount Amount field on a line item was automatically populated to \$0.00. This meant that, for any line item, users had to clear the discount amount field if they wanted to apply a header level discount after adding a line item.

However, some customers implemented Siebel Quotes 6.x to read the Discount Amount field the same whether it was NULL or \$0.00, thereby allowing users to enter a header discount after adding a line item and avoiding the situation illustrated above.

In Release 7.x, this issue is no longer applicable. The Discount Amount field is left as NULL unless a user explicitly enters a value, including \$0.00. Therefore, when a customer upgrades from Release 6.x to Release 7.x, the customer will need to determine whether no change had been made in Siebel Quotes 6.x to Discount Amount (thus letting it be automatically populated to \$0.00) or whether Siebel Quotes 6.x had been modified to read NULL or \$0.00 in Discount Amount as the same. In the event the latter is true, the customer will need to run a Quote script with the following SQL statement applied before beginning the upgrade, that is, before running the configuration utility and applying it to development and production upgrades.

```
update S_QUOTE_ITEM  
  
set  DISCNT_AMT = NULL  
  
where DISCNT_AMT = 0
```

This script will account for the fact that the Discount Amount field is read the same whether it is NULL or \$0.00. Administrators running Siebel Quotes 6.x without modifications to Discount Amount can ignore this script.

Timesheet

If you are upgrading from a pre-7.5 version of Siebel eBusiness Applications, please refer to Siebel SupportWeb for a technical note about Time Sheet Conversion.

Calendar

NOTE: If you are upgrading from Release 7.0.x to Release 7.5, you do not need to perform this procedure.

An optional preupgrade script may be run to prevent duplicate Siebel Sync appointments in Release 7.x calendar. The consequences of running and not running this script are listed below.

If you run the preupgrade script:

- Users will not see duplicate Siebel Sync appointments in Release 7.x.
- Planned Start value will be modified for any record where Release 6.x Planned Start = Creation AND Start Time IS NOT NULL.

If you do not run the preupgrade script:

- Users will see duplicate Siebel Sync appointments in Release 7.x. This will be particularly problematic for repeating appointments.
- Planned start will not be overwritten.

If you wish to run this script, the following SQL statement must be run before the upgrade (this step cannot be performed after the upgrade):

```
update S_EVT_ACT
set TODO_PLAN_START_DT = null
where APPT_START_DT is not null
and APPT_START_TM is not null
and TODO_PLAN_START_DT = CREATED
```

This step will update Siebel Sync imported records TODO_PLAN_START_DT to null.

Preparing Application Tables for Upgrade

DESC_TEXT is a column that resides on two tables, S_CONDITION and S_PROD_BNFT. To broaden platform support, this column will be reduced in length from 250 to 200 during the upgrade from Release 6.x to Release 7.x. Therefore, before you run the upgrade, you must examine these two tables for records which contain more than 200 characters. If you find oversized records, you need to manually reduce the size of these records to 200 characters or less.

The tables that you need to review are:

- S_CONDITION
- S_PROD_BNFT

To prevent loss of data, you need to perform the following procedure before you upgrade to Release 7.x.

To prepare S_CONDITION and S_PROD_BNFT for upgrade

- 1** Review records in S_CONDITION for records that contain more than 200 characters.
- 2** For records which contain more than 200 characters, edit the records so that their row length is less than 200.
- 3** Repeat this procedure for S_PROD_BNFT.

Upgrading the Development Environment

4

This chapter describes the tasks involved in upgrading the Siebel development environment. Do not proceed unless you have completed the preupgrade tasks in [“Preparing the Development Environment Database for Upgrade” on page 60](#) to prepare your development environment data and your repositories for the upgrade.

NOTE: Users who have not customized the Siebel eBusiness Applications, and who therefore do not have a development environment, should proceed directly to [“Production Environment Preupgrade Tasks” on page 245](#).

After you have completed all of the pre–upgrade tasks outlined in the previous chapter, you are ready to perform the actual upgrade of the development environment. [Table 14 on page 104](#) provides of checklist of upgrade tasks for the development environment.

CAUTION: Be sure to uninstall the previous versions of Siebel server and client software before installing the new versions. This should be done to validate proper functioning.

Table 14. Siebel Development Environment Upgrade Tasks

1	Upgrade development workstations. See “Installing Siebel Tools on the Development Workstations” on page 105.
2	Upgrade the Siebel Gateway Name Server and Siebel Servers. See “Upgrading Gateway Name Server and Siebel Servers” on page 106.
3	Install and configure Siebel Database Server Software. See “Installing and Configuring the Siebel Database Server Software” on page 109.
4	Upgrade Siebel database schema. See “Upgrading the Siebel Database Schema” on page 118.
5	Prepare the prior customer repository for merge. See “Preparing the Prior Customer Repository for the Merge” on page 124.
6	Migrate strings, merge labels and fields, and merge templates. See “Migrate Strings, Merge Labels and Fields, and Merge Templates” on page 125.
7	Upgrade copied objects. See “Automatic Upgrade of Copied Objects” on page 127.
8	Perform configuration steps for upgrade inheritance. See “Configuration Steps for Upgrade Inheritance” on page 131.
9	Prepare customized objects for the merge. See “Preparing Customized Objects for the Merge” on page 137.
10	Merge the repository. See “Performing a Repository Merge, Using Siebel Tools” on page 143.
11	Generate EIM temporary columns. See “Generating EIM Temporary Columns” on page 155.
12	Upgrade your custom database schema. See “Upgrading the Custom Database Schema” on page 161.
13	Migrate business component configurations. See “Migrating Custom Business Component Configurations” on page 170.

Installing Siebel Tools on the Development Workstations

The upgrade process requires the Siebel Tools application. If you do not have Siebel Tools, you need to contact Siebel technical support for assistance. Siebel Tools is required even if you have no customizations.

- 1** Remove all but one installation of the previous version of Siebel Tools from all development workstations.
 - For upgrades from version 5.x, double-click the UnInstall icon in the Siebel Tools program folder.
 - For upgrades from 6.x, navigate to the Control Panel, and double-click Add/Remove Programs. Select all but one installation of the previous version of Siebel Tools, then click Add/Remove. Follow the instructions in the InstallShield wizard to uninstall the applications.

CAUTION: Be sure to keep a single installation of the previous version of Siebel Tools. You will need this in order to rename repositories when you upgrade the development environment. Otherwise, you will have to reinstall the earlier version of Siebel Tools in order to perform the production upgrade.

- 2** Install the new Siebel Tools Release 7.x software on all development workstations. For complete instructions on installing Siebel Tools, refer to *Siebel Tools Reference, MidMarket Edition*.

NOTE: If you are uninstalling Siebel eBusiness Applications, Release 6.x.x on the Windows platform, you may experience a failure to uninstall that version due to a bug in InstallShield versions greater than Release 6.0.x.

You will most likely encounter this problem if, on trying to uninstall Siebel eBusiness Applications, you receive a message with the text: *Setup failed to run installation.*

Refer to the chapter on uninstalling Siebel eBusiness Applications in *Siebel Server Installation Guide for Microsoft Windows, MidMarket Edition* for a procedure to prevent this error from occurring. You can also refer to Siebel SupportWeb (<http://ebusiness.siebel.com/supportweb/>).

Upgrading Gateway Name Server and Siebel Servers

You must upgrade all of your development environment servers—Gateway Name Server and Siebel Servers—to the Release 7.x software. If you have multiple servers in your development environment, you will need to perform the upgrade procedure described in this section for all Siebel Servers in your deployment.

Before proceeding, make a copy of your previously completed Upgrade Planning Worksheet. You will need to refer to this during the upgrade process for server names and other installation information. You will also need to refer to *Siebel Server Installation Guide for Microsoft Windows, MidMarket Edition* for server installation instructions.

CAUTION: Do not attempt to install the new version of Siebel eBusiness Applications software without first removing the previous version as instructed. If you attempt this, your installation might not operate correctly.

To upgrade your development environment servers, you must perform the following procedures on each server:

- Stop all Siebel servers and the Gateway Name Server service.
- Uninstall the previous version of the Siebel Server software.
- Uninstall the earlier version of the Gateway Name Server.
- Install the version 7.x Gateway Name Server software.
- Install the version 7.x Siebel Server.

Upgrading the Servers on the Windows Platform

To stop all servers

- Stop all Siebel Servers and Siebel Gateway Name Server service by navigating to Start > Settings > Control Panel > Services.

To uninstall the earlier version of the development Siebel Servers

- 1** If you wish to preserve your installation-specific Siebel Server configuration parameters, record your current configuration.

You will need to reapply that configuration manually after completing the upgrade. Installation-specific parameters will be lost when you uninstall your existing Siebel Servers. When you install the new Siebel Server Release 7.x software, you can reset these parameters manually through the Server Manager.

- 2** Uninstall the previous release of the Siebel Server software.
 - For upgrades from Release 5.x, double-click the UnInstall icon in the Siebel Server program folder.
 - For upgrades from Release 6.x, navigate to the Control Panel, and double-click Add/Remove Programs. Select the previous release of Siebel Server, and then click Add/Remove. Follow the instructions in the InstallShield wizard to uninstall the applications.
- 3** Delete the entire Siebel Server installation directory tree, and then restart the machine.

NOTE: To avoid conflicts with the ODBC drivers used by the new Siebel Server software, remove any older versions of the ODBC drivers. If the vendor provided an uninstall option, use it. If the vendor has not provided an uninstall option, contact the vendor for removal instructions.

To uninstall the earlier version of the Siebel Gateway Name Server

- 1** Uninstall Siebel Gateway Name Server:
 - For upgrades from Release 5.x, double-click the UnInstall icon in the Siebel Gateway Name Server program folder.
 - For upgrades from Release 6.x, navigate to Start > Settings > Control Panel > Add/Remove Programs. Select the previous release of Siebel Gateway Name Server, then click Add/Remove. Follow the instructions in the InstallShield wizard to uninstall the applications.
- 2** Delete the entire Siebel Gateway Name Server installation directory tree, and then restart the machine.

To install the version 7.x Siebel Gateway Name Server software

- Refer to the appropriate chapter on installing the Siebel Gateway Name Server in *Siebel Server Installation Guide for Microsoft Windows, MidMarket Edition*.

To install the version 7.x development environment Siebel Servers

- 1** Make sure that each application server on which a Siebel Server will be installed has the correct versions of all required third-party software products, as documented in *Siebel System Requirements and Supported Platforms*.
- 2** Install the new Siebel Server on all application servers, as described in *Siebel Server Installation Guide for Microsoft Windows, MidMarket Edition*.

Refer to your previously completed copy of the Upgrade Planning Worksheet for server names and other installation information.

Installing and Configuring the Siebel Database Server Software

Configuring the Siebel Database Server for upgrade consists of two sets of tasks:

- [“Installing the Siebel Database Server Software” on page 109](#)
- [“Configuring the Siebel Database Server for Upgrade” on page 110](#)

You will use the Siebel Software Configuration Utility to perform database configuration tasks.

Installing the Siebel Database Server Software

You will need to install the Release 7.x Siebel Database Server software onto one development Siebel Server. The Siebel Database Server must be installed on a Siebel Server that has already been upgraded to Release 7.x.

NOTE: With the Siebel Enterprise Server (SES) installer, when the Siebel Server is chosen, make sure that you choose the Siebel Database Server as well.

CAUTION: In order to configure and execute Siebel Database Server procedures and maintenance scripts, you must have sufficient access to Siebel Server 7.x directories:

You must have READ-WRITE access to `BIN` directories under Siebel Server executables in the `SIEBSRVR_ROOT` directory.

You must have READ-WRITE access to the log directories and upgrade directory.

To install the Release 7.x Siebel Database Server software in your development environment, follow the instructions for installing database server components in *Siebel Server Installation Guide for Microsoft Windows, MidMarket Edition* in the chapter on installing the Siebel Database Server software on your RDBMS. Refer to your completed Upgrade Planning Worksheet for server names and other installation information.

In order to have Siebel Administrator permissions, you must log in with a valid RDBMS user name and password, and this user name must have “Siebel Administrator” responsibility for the default organization.

The default administrator user name is:

SADMIN

If the user SADMIN does not exist or does not have “Siebel Administrator” responsibility for the default organization, you must contact your database administrator to establish this before you proceed with the upgrade.

This attribute can be changed to the login of another “employee” if necessary to meet your business needs.

Installing Additional Languages for a Multilingual Deployment

If your organization deploys internationally and, therefore, requires data to be in multiple languages, you need to install multilingual seed data (for example, lists of views, responsibilities, or system preferences). To do this, you need to add new language packs to your database after you have installed the base language for your database server.

See *Siebel Server Installation Guide for Microsoft Windows, MidMarket Edition* for instructions on installing and configuring multiple Siebel language packs and importing a new language to your repository.

Configuring the Siebel Database Server for Upgrade

The Siebel Software Configuration Utility generates your upgrade configuration file (UCF file):

```
master_<upgrade_option>_<upgrade_type>_<version>.ucf
```

for example,

```
master_upgprep_dev_601.ucf
```

This file will be used to run the upgrade. After your configuration is complete, this file will be created in the *SIEBSVR_ROOT\bin* directory.

The Siebel Software Configuration Utility validates certain parameters, and will not proceed if you enter an invalid parameter. The configuration utility validates the following information:

- That you have renamed the Siebel Repository to Prior Customer Repository
- That tablespaces exist in your database
- That username and password are valid
- That tableowner and tableowner password are valid
- That the language pack is installed
- That the directories chosen exist

To change the language in which the configuration utility runs

When you launch the Siebel Software Configuration Utility, using one of the methods previously mentioned, it launches automatically in the language in which you originally chose to run the SES Installer.

You can change the language in which the configuration utility runs, if desired, from the language chosen during installation. To change the Siebel Software Configuration Utility language, refer to the chapter about using the Configuration Utility in *Siebel Server Installation Guide for Microsoft Windows, MidMarket Edition*.

To configure the Siebel Database Server on the Windows Platform

- 1** Verify that no server tasks are running in the background.

If necessary, stop Siebel Servers and Siebel Gateway Name Server service by navigating to Start > Settings > Control Panel > Services.

- 2** If the Siebel Software Configuration Utility is not already open, launch the utility by selecting Start > Programs > Siebel Enterprise Servers 7.0 > Configure DB Server.

The Siebel Enterprise Parameters: Gateway Server Address screen appears.

- 3 Type the following values as you recorded them in your copy of [Appendix A, “Upgrade Planning Worksheet.”](#)

- **Gateway Server Address.** The alias of your Siebel Gateway Name Server (typically the machine name).
- **Enterprise Server Name.** The name of your Enterprise Server, for example, `siebel`.

To continue, click Next.

The Installation and Configuration Parameters: Siebel Server Directory screen appears.

- 4 Accept the default value displayed in the Siebel Server Directory field, or type an alternate directory path for your configuration. Click Next.

NOTE: This is the `SIEBSVR_ROOT` directory, for example, `D:\sea7xx\siebsvr`.

The Installation and Configuration Parameters: Siebel Database Server Directory screen appears.

- 5 In the Siebel Database Server Directory field, verify that the default directory path displayed is the correct database server installation directory for your configuration. If it is not, use the Browse button to navigate to a different database directory.

NOTE: This is the `DBSRVR_ROOT` directory, for example, `D:\sea7xx\dbsrvr`.

To continue, click Next.

- 6** In the Siebel Database Server Options : RDBMS Platform screen, choose the database platform that you are upgrading.

- **IBM DB2 for Unix and Windows**
- **Microsoft SQL Server**
- **Oracle**

To continue, click Next.

The Siebel Database Server Options: Siebel Database Operation menu appears.

- 7** Choose `Upgrade Database` and click Next.

The Upgrade the Siebel Database: Upgrade Options screen appears.

- 8** Choose `Upgrade Siebel Database Schema (upgrep)` and click Next.

The Upgrade Configuration Information: Environment Type screen appears.

- 9** Choose `Development` to upgrade your development environment.

To continue, click Next.

The Upgrade Configuration Information: Current Siebel Version screen appears.

- 10** Choose the base version of Siebel from which you are upgrading.

To continue, click Next.

The Installation and Configuration Parameters: Language Selection screen appears.

NOTE: The language selection screen will not appear if you have only one language installed, because that language is automatically defined as the primary language. In this case, you will continue with [Step 12](#).

- 11 Choose the language which was the primary language of your prior environment.

To continue, click Next.

NOTE: If you cannot continue, then you selected a language for which you do not have a language pack. You need to reselect the primary language used in your prior environment, and then click Next to continue.

The next Installation and Configuration Parameters: ODBC Data Source Name screen appears.

- 12 Accept the name of the ODBC data source displayed for verification purposes (for example, `SiebSrvr_siebel`), or enter a different data source name.

NOTE: The data source is created automatically by the Siebel Server installation, using the format `SiebSrvr_EnterpriseName`. To find the name of your ODBC data source, navigate to Start > Settings > Control Panel > Administrative Tools > Data Source (ODBC). Click the System DSN tab and you will find the name of your ODBC data source.

To continue, click Next.

The Installation and Configuration Parameters: Database User Name screen appears.

- 13 Type the source user name and password for your database:
 - a **Database User Name.** User name of the Siebel administrator, for example, `sadmin`.
 - b **Database Password.** Password for the Siebel administrator, for example, `sadmin`.
 - c **Database Password (confirm).** Retype the password for confirmation.

To continue, click Next.

NOTE: The following Installation and Configuration Parameters screens are platform-specific. Which screens appear next will depend on the database platform that you are upgrading.

- 14** In the Installation and Configuration Parameters screens that appear next, complete each field with the values that you recorded on your Upgrade Planning Worksheet. The default values for each RDBMS are listed in the table below.

NOTE: Use underscores rather than spaces; these values are case-sensitive.

Field	Value
Database Table Owner	<p>This is the account that will own the Siebel objects, for example, <code>siebel</code>.</p> <ul style="list-style-type: none"> ■ DB2 UDB: Tableowner. ■ Microsoft SQL Server: Database Owner Login (this is the login for the owner of the database, not necessarily the default owner of the database in DBO). ■ Oracle: Tableowner.
Database Table Owner Password	<ul style="list-style-type: none"> ■ DB2 UDB: Tableowner password. ■ Microsoft SQL Server: the password for Database Owner Login (this is the login for the owner of the database, not necessarily the default owner of the database in DBO). ■ Oracle: Tableowner password.
Index Space	<ul style="list-style-type: none"> ■ DB2 UDB: The name you give to your 4-KB index space for tables. ■ Oracle: The name you gave to your index area.
4KB Table Space	<ul style="list-style-type: none"> ■ DB2 UDB: The name you gave to your 4-KB table space. <p>NOTE: Be sure to use underscores rather than spaces.</p>
8KB Table Space (DB2 UDB upgrades from 5.x only)	<ul style="list-style-type: none"> ■ DB2 UDB: The name you gave to your 8-KB table space.
16KB Table Space (DB2 UDB only)	<ul style="list-style-type: none"> ■ DB2 UDB: 16 KB table space name.
32KB Table Space (DB2 UDB only)	<ul style="list-style-type: none"> ■ DB2 UDB: 32 KB table space name.
Table Space Name	<ul style="list-style-type: none"> ■ Oracle: The name you gave to your data area.

After you type the value for each screen, to continue, click Next.

- 15** For all platforms, in the Upgrade Configuration Information: Database Server OS screen, choose the platform on which your database server runs.

To continue, click Next.

The Configuration Parameter Review screen appears.

- 16** Review the configuration values you entered on the previous Configuration Utility screens against the values that you recorded in your copy of [Appendix A, “Upgrade Planning Worksheet.”](#)

NOTE: Passwords are encrypted and will not appear in plain text either in the user interface or in the upgrade configuration files (UCF files). After a password is entered, it will always appear in encrypted form. If you need to use another password, you must rerun the configuration utility.

- If you need to go back to make changes, click Previous to back out until you reach the screen with the parameter you need to change. Enter the valid parameter, and then click Next until you reach the Configuration Parameter Review screen again.
 - To accept the values you input with no changes, click Finish.
- 17** A message box appears, prompting you to decide if you want to apply the configuration now or later.
- To apply the configuration now, click OK.

The Siebel Upgrade Wizard appears. To begin the upgrade of your Siebel Database Schema, click OK. A check mark will appear beside each item as it is completed. When the status bar registers that the upgrade process is complete, click OK to exit the Siebel Upgrade Wizard.

At this stage, you have finished upgrading your Siebel Database Schema and are ready to review the log files for errors. Read the instructions below, and then see [“Reviewing the Upgrade Log Files” on page 119.](#)

- To apply the configuration later, click Cancel.

When you are ready to apply your configuration, follow the procedure: [“To apply your Siebel Software configuration later” on page 117.](#)

CAUTION: If you cancel or abort the upgrade process at any point, ask your database administrator to terminate the upgrade process connection to the database.

During the upgrade, the only active connections to the database should be those required for the upgrade. At no time should there be any online users connected using the Siebel eBusiness Application.

If errors are encountered during the upgrade process, it will stop at that point. If your upgrade stops due to an error, you must carefully review several log files to make sure that your upgrade has completed successfully up to that point, and fix the error that stopped the upgrade. See [“Reviewing the Upgrade Log Files” on page 119.](#) Once you have corrected the error, you may restart the upgrade, and it will continue from the last step that completed successfully. For details on how to restart, see [“Reviewing the Upgrade Log Files” on page 119.](#)

To apply your Siebel Software configuration later

- 1 If you want to apply the configuration later, first you need to review the upgrade configuration file (UCF file) under the *SIEBEL_ROOT\siebsrvr\bin* directory:

```
master_<upgrade_option>_<upgrade_type>_<version>.ucf
```

for example,

```
master_upgrep_dev_601.ucf
```

- 2 After you have reviewed the UCF file, to apply the configuration, enter the following command in the DOS command prompt:

```
SIEBEL_ROOT\bin\siebupg.exe /m master_<upgrade_option>_<upgrade_type>_<version>.ucf
```

for example,

```
SIEBEL_ROOT\bin\siebupg.exe /m master_upgrep_dev_601.ucf
```

Upgrading the Siebel Database Schema

When you completed the previous procedure, “[Configuring the Siebel Database Server for Upgrade](#)” on page 110, the Siebel Software Configuration Utility automatically launched the Siebel Upgrade Wizard.

During the development environment upgrade of Siebel Database Schema, the Siebel Upgrade Wizard performs the following major tasks:

- Upgrades the Siebel database schema to the new version
- Migrates application data to the new data model structure
- Upgrades Siebel seed data
- Loads three new repositories:
 - Prior standard repository (version of 5.x or 6.x)
 - New standard repository (version 7.x)
 - New customer repository (version 7.x) which, after the merge, will become the customized 7.x repository.

The Siebel Upgrade Wizard is restartable at most stages within the upgrade process. For information about restartability, see “[Restarting the Upgrade](#)” on page 168.

CAUTION: If problems with your environment prevent the upgrade from restarting, you need to restore the database from the prior base version (the version which you are upgrading from). For example, environment problems may occur when table creation fails due to a database problem (insufficient storage or network problems), which cause subsequent upgrade steps to fail. If you need to restore your database and restart the upgrade, you should delete or store the upgrade log files that were generated in the SiebServ\Log directory to a zip file. You should also delete the log file, `state.log`, from the `SIEBSVR_ROOT\upgrade\driver_upgrep_dev_version` directory.

Reviewing the Upgrade Log Files

After an upgrade, depending on the trace level that you set, several log files are generated. Review these log files for unacceptable errors.

- If the upgrade completed successfully, there are several log files that you can safely ignore:
 - `sw_cfg_xxx.log`
 - `siebel.log`
 - Any other log file that existed prior to the start of the upgrade
- If you encountered an error during the upgrade, review the log files generated by the upgrade wizard, such as `UpgWiz.log`, `UpgWiz_01.log` (the name of the log file will increment for subsequent log files) within the `SIEBEL_ROOT\siebsrvr\LOG` directory. Subsequent log files are created when the upgrade wizard encounters a problem and the user attempts a retry.

Review the end of the `UpgWiz.log` file for details about the latest failure. If the step that failed was not a native SQL step (which would be listed in `UpgWiz.log`), then it occurred as part of an external utility, for which you need to review a corresponding log file, identified by the `/L` parameter.

NOTE: Archive your log files in case future analysis is required by technical support. (Preserve the date and time of the log files, because a copy of the files will overwrite the date and time stamp.)

The log files may include errors that are expected and benign. Compare any error messages found in the log files to the sample error messages in the `errors.rtf` file located in the `DBSRVR_ROOT\PLATFORM` directory.

To review the log files for unacceptable errors

- 1 Print the `errors.rtf` file. This file is located in the installation subdirectory for your database platform.
- 2 Sort the files in the `SIEBEL_ROOT\LOG` directory according to date, with the most recent files on top.
 - The log files are identified by the `.log` extension.

Only one of each type of error occurring in a particular log file appears in the `errors.rtf` file.
 - If a log file is not listed in the `errors.rtf` file, then there are no acceptable error messages for that log file.
- 3 Open each log file, starting with the earliest, and perform a global search for errors, which are either tagged with the word “error” or enclosed in square brackets “[...]”.

NOTE: It is important that you start with the earliest log file. It will shorten your research time if you find unacceptable errors in an early log file.

- 4 For each error found, look for that error in the list of acceptable errors documented in the `errors.rtf` file located in the `siebsrvr_root\log` directory.
 - If you locate the error in the `errors.rtf` file, it is acceptable and no action is required. Continue to review the errors found in the log file.
 - If you find an error that is not listed in the `errors.rtf` file, it is unacceptable. You must correct the condition that caused the error before you rerun the Upgrade Wizard.

Report the error to Siebel Technical Support. *Do not proceed with the upgrade.*

- 5 Repeat the previous step for each log file.

CAUTION: Do not proceed with the upgrade until unacceptable errors have been corrected. If you cannot correct the error, contact Siebel Technical Support or Professional Services to report the error in detail.

Do not review error numbers only, because these may have changed following installation of a new driver version. Instead, compare the actual error descriptions to find out which are acceptable errors for this platform.

Although other errors are rarely encountered, this review is critical. Certain errors, such as a failure to create indexes, may result in performance problems or anomalous behavior in Siebel eBusiness Applications.

Troubleshooting the Siebel Database Schema Upgrade

Typical problems that may occur at this stage could result from a lack of storage space or insufficient user privileges.

Recovering from a Failed Siebel Database Schema Upgrade

If the repository upgrade fails due to insufficient space allocated on the database, you must complete the following procedures.

To recover from a failed Siebel database schema upgrade

- 1** Back up your complete set of log files, from the beginning of the upgrade process to the point at which it stopped, to another directory.
- 2** Read the upgwiz.log file and associated log file to determine the failure. See [“Reviewing the Upgrade Log Files.”](#)
- 3** Take the necessary corrective action. Depending on the errors that you find, you may need to ask your DBA to extend the database.
- 4** Resume the upgrade wizard. It will continue from the point at which it failed.

NOTE: To resume the upgrade wizard, type the following command at the Windows DOS command prompt from the siebsrvr_root/BIN directory:
`siebugp.exe /m master_<upgrade_option>_<upgrade_type>_<version>.ucf`

For example, to restart a development upgrade from Release 6.0.1 to Release 7.x, type: `siebugp /m master_upgrep_dev_601.ucf`.

Restarting the Upgrade

The Siebel Upgrade Wizard is restartable at most stages within the upgrade process. If the Siebel Upgrade Wizard encounters an error during the upgrade process, it will stop at that point. If your upgrade stops due to an error, you must carefully review the relevant log files to make sure that your upgrade has completed successfully up to that point. Once you have verified this and resolved the failure, you may restart the upgrade. The upgrade will continue from the last step that completed successfully.

CAUTION: Before you restart the upgrade (after any break in the upgrade process) back up your complete set of log files, from the beginning of the process to the point at which it stopped, to another directory. This will maintain a complete record of your log files, and prevent your previous log files from being overwritten, which could prevent accurate diagnosis of the reason for the break in the upgrade.

Launching the Siebel Upgrade Wizard

To manually launch the Siebel Upgrade Wizard from the command prompt, or to restart the upgrade

- 1 Type the following command at the Windows DOS command prompt from the `SIEBEL_ROOT\siebsrvr_root\BIN` directory:

```
siebug.exe /m master_<upgrade_option>_<upgrade_type>_<version>.ucf
```

For example, to restart a development upgrade from Release 6.0.1 to Release 7.x, type the following command:

```
siebug.exe /m master_upgrep_dev_601.ucf.
```

The Siebel Upgrade Wizard screen appears, displaying the items to be completed during this phase.

- 2 To begin the upgrade of your repository, click OK.
A check mark will appear beside each item as it is completed.
- 3 When the status bar registers that the upgrade process is complete, click OK to exit the Siebel Upgrade Wizard.

You have finished upgrading your repository and are ready to review the log files for errors. See [“Reviewing the Upgrade Log Files” on page 119](#).

Taking a Backup of the Database Repository

Back up your database repository after a successful upgrade of the Siebel Database Schema.

NOTE: This backup will enable you to restore your pre-merge database if you need to recover from a failed repository merge. See [“Recovering from a Failed Merge” on page 155](#).

Adding New License Keys

With the new Release of Siebel eBusiness Applications, you received one or more license keys that must be added to the development database. You must add all of the new license keys to enable the new release of the Siebel eBusiness Applications, which you will use in the next steps of the upgrade process.

To add new license keys

- 1** Start Siebel Tools version 7.x from a development workstation and log on to the database server as the Siebel administrator.
- 2** Add your new license keys.

Preparing the Prior Customer Repository for the Merge

You will perform the several procedures to prepare the Prior Customer Repository for the repository merge.

- **Run the Repository Preparation Wizard.** The Repository Preparation Wizard needs to be run on the Prior Customer Repository before you proceed with the repository merge. This utility is invoked from Siebel Tools, and will be used to perform the following procedures:

- **Migrate Strings**
- **Merge Labels and Fields**
- **Merge Templates**

See [“Migrate Strings, Merge Labels and Fields, and Merge Templates.”](#)

CAUTION: If you are upgrading from Release 7.0.x to Release 7.5, do not run the Repository Preparation Wizard. You already completed this procedure during your upgrade to Release 7.0.x.

- **Upgrade Copied Objects.** The ability to upgrade copied objects is a new feature for Release 7.x upgrades. Copied objects are the customized objects from prior versions of Siebel software. In prior upgrades, to Release 6.x, only Siebel objects were upgraded. In Release 7.x upgrades, your custom objects will be upgraded if they have an upgrade ancestor specified. See [“Automatic Upgrade of Copied Objects”](#) on page 127.

Migrate Strings, Merge Labels and Fields, and Merge Templates

Before you run the repository merge, you need to run the Repository Preparation Wizard from Siebel Tools to perform the following operations:

- **Migrate strings.** Release 7.x supports locale strings in object-specific tables. In order to move the S_MSG data in your environment to the new locale table structure in Release 7.x, you need to run the Migrate Strings utility.

NOTE: Migrate strings for each language that was supported in your prior repository.

- **Merge labels and fields.** Release 7.x merges labels with controls so that the label is now recognized as the caption property of the control. Before you run the repository merge, you need to run the Repository Preparation Wizard to merge labels, controls and fields from the previous version based upon form applet layout.
- **Merge Web templates.** Release 7.x merges web templates in Siebel Tools to only use Base and Edit/Query/New for the majority of the out of the box applets. Previous releases of Siebel used four separate templates to deliver the same functionality. Before you run the repository merge, you need to run the Merge Templates utility to merge your applet web templates.

The Repository Preparation Wizard will prompt you to confirm that you completed the following steps before it will proceed with the repository merge.

To migrate strings, merge labels and fields, and merge applet web templates

- 1 In Siebel Tools, navigate to the Tools menu, and then choose Tools > Upgrade > Prepare Repository.

Choose the Repository that you wish to prepare.

The String Migration window appears.

- 2 In the String Migration window:
 - a Select the language for which you wish to migrate strings.
 - b To log migrated strings, click in the check box beside *Log migrated strings*, then click the browse button to specify the log file.

To continue, click Next. The Merge Labels and Fields window appears.

3 In the Merge Labels and Fields window:

a Specify the input file by clicking the browse button.

NOTE: The utility will operate on every form applet except those specified in the input file. The default input file is `applets.txt`. Modify the input file only if you have additional applets that should not be merged.

b Specify the location of web templates.

To continue, click Next. The Merge Applet Web Templates window appears.

4 In the Merge Applet Web Templates window, click the browse button to specify the input file, then click Next.

NOTE: The utility will operate on every form applet except those specified in the input file. The default input file is `applets.txt`. Modify the input file only if you have additional applets that should not be merged.

The Merge Applet Web Templates window appears. Click OK to confirm that you want to proceed.

The wizard will prepare your Prior Customer Repository for the merge.

Automatic Upgrade of Copied Objects

Siebel Tools allows copied objects to inherit some of the behavior of their ancestors, which makes it easier to upgrade Siebel applications, reduces the time and cost of adjusting an application after an upgrade, and also supports parallel development by allowing some frequently used objects to be copied.

Certain repository objects that are copied during configuration can be upgraded with a new property called Upgrade Ancestor that stores the name of the ancestor object. This allows copied objects to be upgraded in the same way as the ancestor objects from which they were copied. Thus when you copy an existing object, you can specify its upgrade ancestor; during an upgrade the copied object will be upgraded the same way as the original. This feature is available only for objects of type Applet, Business Component, Report, and Integration Object.

NOTE: Use of the Upgrade Inheritance feature will slow the performance of the repository merge.

Upgrade Inheritance functionality:

- The Upgrade Ancestor property stores the name of the ancestor object (that is, the one from which the current object was copied).
- If the Upgrade Ancestor property is not null, you can upgrade the copied object as if it were the ancestor object.
- No special action is taken during import even if the “Upgrade Ancestor” property is specified, because this property is specific to Application upgrades. However, imported objects can have the Upgrade Ancestor property set. When the next application upgrade is done, the property is taken into account.
- Inheritance does not apply to patch files. The “Upgrade Inheritance” property is applicable only during application upgrades. It is not taken into account during the application of a patch and no action is performed based on this property.

- During the merge, the newly created objects are given all the changes corresponding to its ancestor. Objects with the Upgrade Ancestor property include:
 - Applets
 - Business Components
 - Integration Objects
 - Reports

Basically, you can create a copy of an existing object (applets, business components, integration objects, and reports) and specify an Upgrade Ancestor.

Upgrade Inheritance Scenario

For example, you may want to make a copy of the Account List Applet and call it the Premium Account List Applet. This new applet may differ from the original one in that it has a special search specification that is displayed only in those accounts that are considered premium accounts. In a subsequent release, Siebel eBusiness Applications may include additional standard list columns to the Account List Applet. During an application upgrade, your Account List applet and the Premium Account List Applet will retain the configuration changes you made. However, both applets will receive the new standard list columns added in the new version because of Upgrade Inheritance functionality. Without this new feature, the copied applet would not receive the new list columns during the upgrade process.

How Enhancements Are Applied During an Upgrade

During upgrades, it is very common that objects in the repository are changed. For example, an applet might have a few list columns added or a business component might have some fields and a multi-value link added. To do this, the objects that need to be changed during the upgrade are recognized by their Name property. For example, you would query the repository for the Account BC and add the necessary new items to it. If you did not have the Upgrade Inheritance feature and the Account BC had been copied as Acme Account, you would not recognize the new BC as a copy of the Account BC and would not add the required changes to the copy during the upgrade. These additions might be minor, but often these omissions can cause numerous application errors after the upgrade and can be time consuming to detect and correct.

During an upgrade, the Upgrade Inheritance feature makes sure that copied objects receive the same changes that are applied to the object from which they were copied. This is done automatically by the upgrade utility, and there is no manual step involved except for specifying the property.

NOTE: This functionality is applied only to the following object types: business component, applet, integration object, and report.

Choosing an Upgrade Ancestor

When choosing an upgrade ancestor for an object, the picklist of objects displayed from which you can choose varies depending on the object type. The picklist has the following constraints for each object type:

- Applets that have the following characteristics:
 - Table is the same as the current applet buscomp
 - Class is the same as the current applet class
 - Upgrade Ancestor is null
 - Available out of the box—Siebel supplied object: shipped as part of the Standard Siebel repository
- Reports that have the following characteristics:
 - Buscomp is the same as the current report buscomp
 - Class is the same as the current report class
 - Upgrade Ancestor is null
 - Available out of the box; Siebel supplied object
- Business Components that have the following characteristics:
 - Bus Comp is the same as the current business component
 - Class is the same as the current business component
 - Upgrade Ancestor is null
 - Available out of the box; Siebel supplied object

- Integration Objects that have the following characteristics:
 - Base Object Type is the same as the current Base Object Type
 - Business Object is the same as the current business object
 - Upgrade Ancestor is null
 - Available out of the box; Siebel supplied object

The constraint requires that these picklists show only standard objects; this can be relaxed by setting a flag found in View > Options > General. This may be appropriate for customers that will use the inheritance feature to support distributed development. Relaxing this constraint does not change the fact that ancestor objects must be found in the New Standard repository in order to be applied to their descendants during a merge.

NOTE: If an object that does not exist in the 7.5 New Standard repository is specified as an ancestor object, you will receive error messages during the repository merge process in the merge.txt file. These errors are acceptable. However, you may want to manually update the descendant objects of the ancestor object because these objects were not updated with the characteristics of the ancestor object during the merge.

Repository Location of the Upgrade Ancestor

During the application upgrade, the contents of three repositories are compared to produce the final, postupgrade repository, which contains both the customizations made by the customer as well as any enhancements added in this release during the upgrade. The three repositories compared are the following:

- Prior Standard Repository: Ancestor Repository.
- Prior Customer Repository: Ancestor Repository that has been customized by the client.
- New Standard Repository: New Siebel Repository.

The Upgrade Ancestor object of a copied object must exist in the New Standard repository in order for any enhancements to be applied to descendants during the merge.

- **New Customer Repository:** New Siebel Repository customized by the client. Before the merge, this is the New Siebel Repository; after the merge, it is a Release 7.5 repository, which contains customizations made by the customer in the previous release, identified in the Prior Customer Repository.

The outcome of the upgrade ancestor only affects the New Customer Repository.

Configuration Steps for Upgrade Inheritance

After you have decided it is necessary to create a copy of an object in the repository, it is easy to specify the parent object from which the copy was created. After creating the copied object, specify the parent object name in the Upgrade Ancestor property of the copied object. This is what allows the copied object to be recognized as a copy during the application upgrade, and it is changed along with its parent object.

NOTE: You must manually populate this property since it is not automatically populated for you when you copy an object. Remember that this property can only be populated if the copied object is an applet, business component, integration object, or report object type.

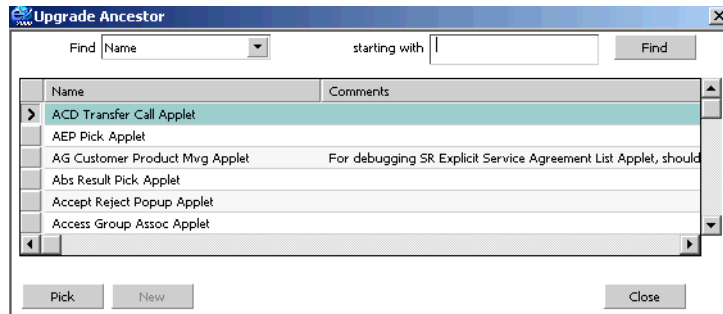
To copy an applet, business component, integration object, or report object and children

- 1** Select the Object type (Applet, Business Component, Integration Object, or Report) in the Object Explorer.
- 2** Select an entry in the Object list applet.
- 3** Choose Edit > Copy Record to create a copy of this record.
- 4** In the new record, fill in a new name in the Name field.

Upgrading the Development Environment

Preparing the Prior Customer Repository for the Merge

- 5 In the Upgrade Ancestor field, select a value from the Upgrade Ancestor dialog box.



The picklist shows all the other business components, applets, integration objects, and reports in the repository.

The following figure shows that the Access Control Employee BC is the Upgrade Ancestor for the copied BC.

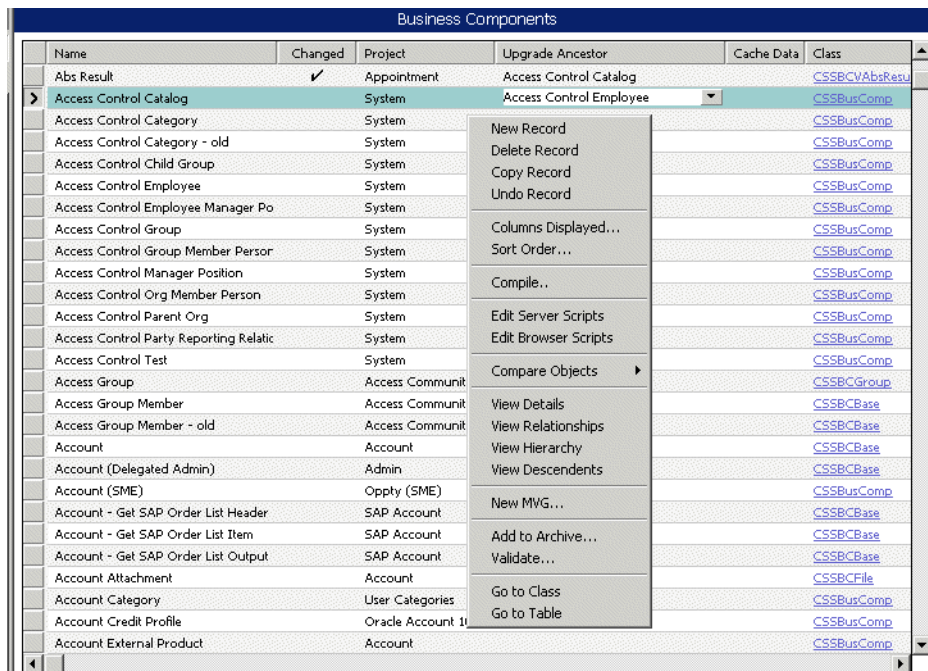
Business Components					
Name	Changed	Project	Upgrade Ancestor	Cache Data	Class
Abs Result	✓	Appointment	Access Control Catalog		CSSBCV/AbsResu
Access Control Catalog		System	Access Control Employee		CSSBusComp
Access Control Category		System			CSSBusComp
Access Control Category - old		System			CSSBusComp
Access Control Child Group		System			CSSBusComp
Access Control Employee		System			CSSBusComp
Access Control Employee Manager Po		System			CSSBusComp
Access Control Group		System			CSSBusComp
Access Control Group Member Person		System			CSSBusComp
Access Control Manager Position		System			CSSBusComp
Access Control Org Member Person		System			CSSBusComp
Access Control Parent Org		System			CSSBusComp
Access Control Party Reporting Relatic		System			CSSBusComp
Access Control Test		System			CSSBusComp
Access Group		Access Community			CSSBCGroup
Access Group Member		Access Community			CSSBCBase
Access Group Member - old		Access Community			CSSBCBase
Account		Account			CSSBCBase
Account (Delegated Admin)		Admin			CSSBCBase
Account (SME)		Oppty (SME)			CSSBusComp
Account - Get SAP Order List Header		SAP Account			CSSBCBase
Account - Get SAP Order List Item		SAP Account			CSSBCBase
Account - Get SAP Order List Output		SAP Account			CSSBCBase
Account Attachment		Account			CSSBCFile
Account Category		User Categories			CSSBusComp
Account Credit Profile		Oracle Account 10.7			CSSBusComp
Account External Product		Account			CSSBusComp

To view all descendants or copies of an object

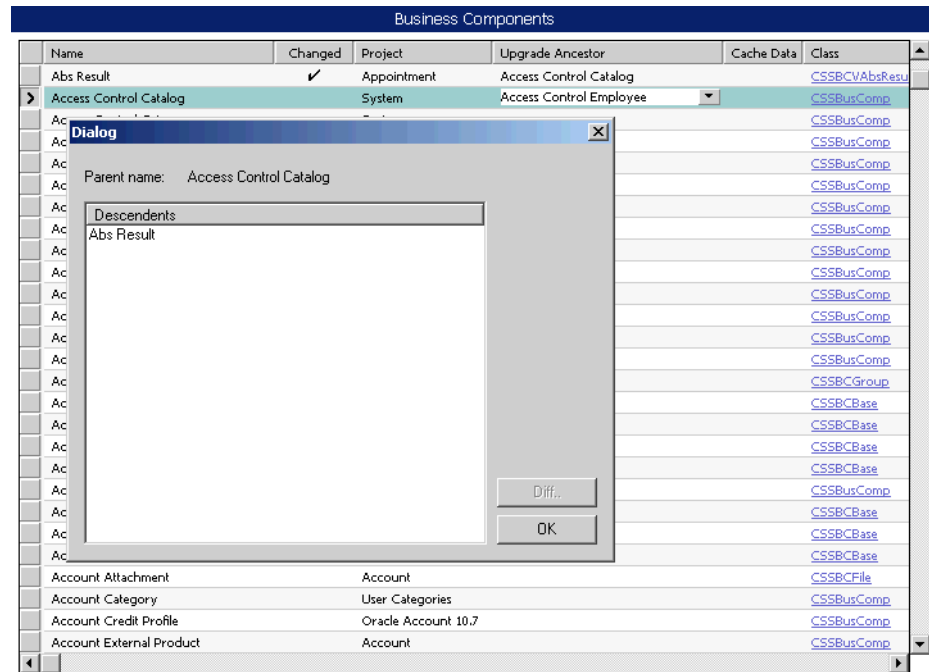
- 1 Right-click on an object.

A dialog box appears.

- 2 Select View Descendants from the dialog box.



The following dialog box appears showing the Parent Name and Descendants.



UI objects should be copied if the look-and-feel of the application will change significantly or if there is a difference needed between two objects (that is, different search specifications on two applets). Business Components should only be copied after all other configuration approaches have been exhausted, and copying is clearly the only solution. The issues involving repository maintenance and specialized classes still are present with copied BCs. Upgrade Inheritance functionality allows certain copied objects to be upgraded and inherit the same characteristics that the parent object possessed. This avoids postupgrade errors and configuration problems.

Propagating Changes from a Parent to Descendants

It is possible to propagate changes that are made to the parent to the descendants of that parent. This is accomplished with the new Object Comparison and Synchronization feature, which allows two objects to be compared, and the differences between the two objects to be applied to one another to keep them synchronized.

NOTE: The Object Comparison and Synchronization feature can be used to compare any two objects. The objects do not need to have a parent-descendant relationship.

Preparing Customized Objects for the Merge

If you have customized objects on your MidMarket Release 5.x application repository, you must perform several tasks on the Prior Customer Repository before you perform the merge:

- Identify Changed Objects
- Rename Changed Objects

If you do not follow this procedure, you may have to reapply the customizations made to Release 5.x products to Release 7.x products or repeat the repository merge process.

NOTE: You should only use this procedure if you have performed customizations to Siebel MidMarket Edition application Release 5.x. This does not need to be performed for purely custom objects such as a new Applet or View created in the Release 5.x product.

Identifying Changed Objects

You need to identify Release 5.x objects with the “SWG” suffix that have been customized or changed. You will need to rename these custom objects to have the same names as the corresponding Release 7.x objects. An installation of Siebel Tools Release 7.x is required in order to identify changed objects.

If you find that there are no custom objects in Release 5.x, then proceed directly to [“Performing a Repository Merge, Using Siebel Tools” on page 143](#).

To identify objects in the Prior Customer Repository Release 5.x that need to be renamed

- 1** Launch Siebel Tools Release 7.x, and log in.
- 2** Choose File > Open Repository. Select Prior Customer Repository > Open.
- 3** Choose View > Options.

The Development Tools Options screen appears.

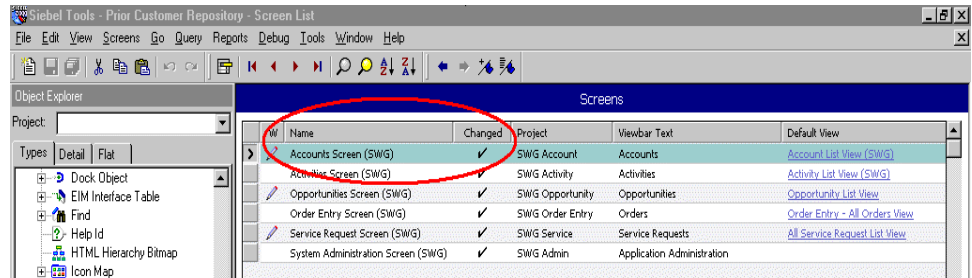
- 4** Select the General tab, then in the Date drop down list, select a date which occurred before the implementation and customizations were started. The changed or customized objects will appear with check marks in the Changed column.
- 5** Select an object-type in the Object Explorer window, for example Applet, View, or Screen. A list of all the objects for this object type will display in the main window.
- 6** Refine the search to look for all the objects of this particular object type that were customized in Siebel MidMarket Edition Release 5.x and have "SWG" as a suffix. Follow this step for each object type (Applet, View, and Screen).
 - a** Select Query > New Query, and type *SWG* in the Name field and a check mark in the Changed column.
 - b** Run this query by selecting Query > Execute Query.
 - c** Export a list of these objects, and create a master list of all the objects that will be used to rename objects.

CAUTION: All objects relating to the Team Forecast View (SWG) and the Personal Forecast View (SWG) must not be renamed because they do not exist in the Release 7.x repository without the "(SWG)" suffix.

The following objects must not be renamed:

- ☐ Personal Forecast View (SWG)
- ☐ Team Forecast View (SWG)
- ☐ Personal Forecast Analysis Chart Applet (SWG)
- ☐ Personal Forecast List Applet (SWG)
- ☐ Team Forecast Analysis Chart Applet (SWG)

- Team Forecast List Applet (SWG)



Create a reference list of the names of changed objects

Once the changed objects are identified in Prior Customer Repository, export those objects into a Comma Separated File (CSV file) and use this list of objects as a reference during the Upgrade Process.

- In Siebel Tools Release 7.x, select File > Export.
- In the Export window, select the following options (as depicted in [Figure 5 on page 140](#)).

Amount to export: All Rows in Current Query

Columns to export: Selected Columns.
Click Choose. In the Columns Displayed dialog box, select Name as the only column to be exported.

Output Format: CSV (Comma Separated Value)

Output file: File to be saved as the master file

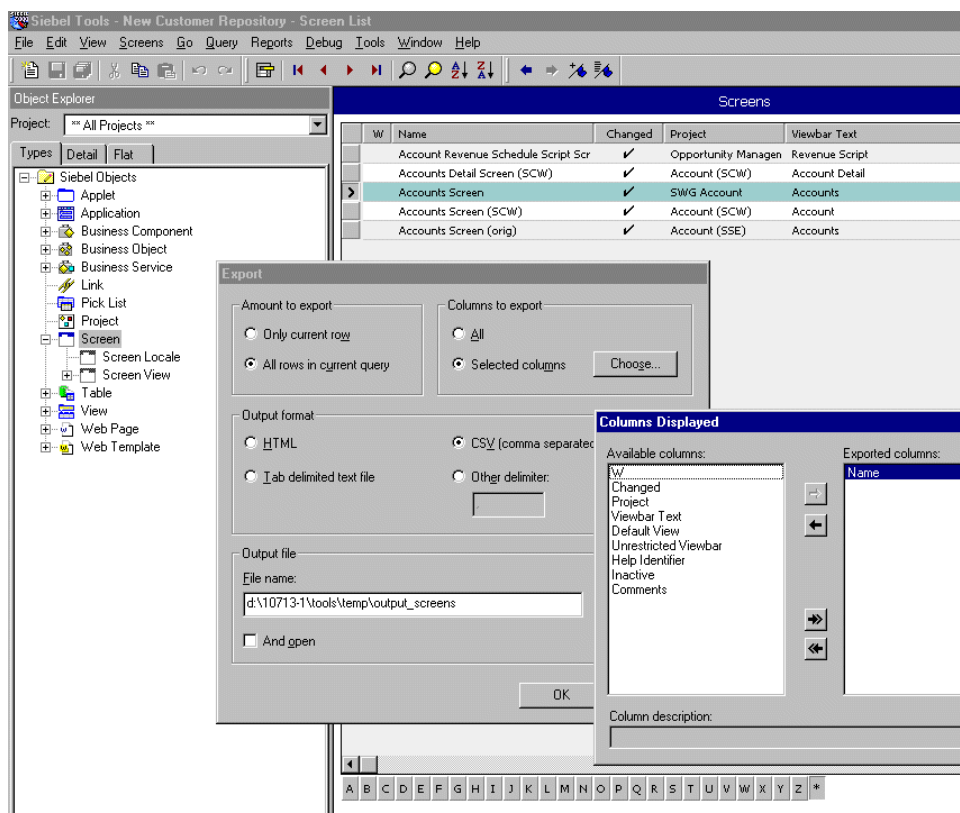


Figure 5. Export a List of Changed Objects With SWG Suffix in the Prior Customer Repository

Renaming Changed Objects

Now that you have a list of changed objects, you need to rename these objects in the Release 5.x Prior Customer Repository and Prior Standard Repository. An installation of Siebel Tools Release 7.x is required in order to perform the following tasks:

- Rename objects in Prior Customer Repository.
- Rename objects in the Prior Standard Repository.

To rename objects in Prior Customer Repository

- 1** Launch the Siebel Tools Release 7.x, and log in. Choose File > Open Repository. Select Prior Customer Repository > Open.
- 2** For every object in the master list (Comma Separated File) that you exported in [“Identifying Changed Objects” on page 137](#), search for objects in the Prior Customer Repository using the Name field in Siebel Tools Release 7.x. Use a wildcard (*) before and after the object name in the Name field for the search.

For Example: Select Query > New Query. Enter *Account Activity List Applet* in the Name field. Select Query > Execute Query to run the query.
- 3** Two objects with the same search string will display in the window, one with a (SWG) suffix and the other one without a suffix.
- 4** Select Tools > Lock Project to make sure the project is locked. Select the object without the SWG suffix, add type the suffix (orig) at the end of the object name to differentiate it from the object name below it.
- 5** Select the object with the swg suffix, and delete (SWG) from the object name.
- 6** Repeat steps 2 through 5 for each object in the master list.

To rename objects in Prior Standard Repository

- 1** Launch the Siebel Tools Release 7.x, and log in. Choose File > Open Repository. Select Prior Standard Repository > Open.
- 2** For every object in the master list (Comma Separated File) that you exported in [“Identifying Changed Objects” on page 137](#), search for objects in the Prior Standard Repository using the Name field in Siebel Tools Release 7.x. Use a wildcard (*) before and after the object name in the Name field for the search.

For Example: Select Query > New Query. Enter *Account Activity List Applet* in the Name field. Select Query > Execute Query to run the query.
- 3** Two objects with the same search string will display in the window, one with a (SWG) suffix and the other one without a suffix.
- 4** Select Tools > Lock Project to make sure the project is locked. Select the object without a suffix, and type the suffix (orig) at the end of the object name to differentiate it from the object name below it.
- 5** Select the object with the SWG suffix, and delete (SWG) from the object name.
- 6** Repeat [Step 2](#) through [Step 5](#) for each object in the master list.

You are now ready to perform the repository merge.

Performing a Repository Merge, Using Siebel Tools

During the repository merge, objects from the Prior Siebel Repository, Prior Customer Repository, and New Siebel Repository are compared by name to identify the total set of object differences. The process also determines how conflicts between repository changes will be resolved as they are merged into the New Customer Repository.

There are three basic categories of object differences:

- New
- Deleted
- Modified

The repository merge executes the following processing steps to identify object differences:

- **New or deleted objects.** Identify objects that the customer has added by comparing their names in the Prior Customer Repository with the Prior Siebel Repository.

All new customer objects are carried over from the Prior Customer Repository to the New Customer Repository. The repository merge typically avoids deletion of objects. Most of the objects that are deleted in the Prior Customer Repository will reappear after the merge. The merge does this to avoid accidental deletion of objects which may be required. It does, however, allow deletion of specific types of objects. Such objects will be deleted from the New Customer Repository during the merge.

Objects of the following types will be deleted from the New Customer Repository:

- | | |
|---------------|----------------------------|
| ■ Control | ■ Chart |
| ■ List Column | ■ Applet Web Template Item |
| ■ Page Tab | ■ View Web Template Item |

- **Objects with altered attributes.** Identifies objects that exist in both the Prior Customer Repository and the New Siebel Repository, and compares the attributes of each object to determine if they have been modified. Attribute comparisons are of interest only for those attributes which were changed by the customer.

If an object attribute was altered in the Prior Customer Repository, but not in the New Siebel Repository, the customer's attribute value will be merged into the New Customer Repository.

A conflict occurs, however, if an object attribute was altered in both the Prior Customer Repository and the New Siebel Repository, in which case the values in all three repositories would be different. In this event, the repository merge process uses the setting of the object attribute's `StandardWins` flag to determine how to resolve the conflict. If this is set to `Y`, the attribute value from the New Siebel Repository will be used; if this is set to `N`, the attribute value from the Prior Customer Repository is used. Conflict resolutions can be overridden for each object attribute in the New Customer Repository. For examples of conflict resolution, see [“Examples of Conflict Resolution” on page 153](#).

Improving Performance of the Merge

There are several ways in which you can reduce the time required to complete the merge.

- 1 Optimize the machine on which you are running the merge.
 - Use a workstation with a minimum of 512 megabytes (MB) of RAM.
 - Allocate at least 2 GB of virtual memory, and a 2 GB page file. If the amount of virtual memory on the system is too low, performance will degrade significantly.
 - Close all other applications.
 - Defragment the disk. Fragmentation significantly affects system performance.

- On the workstation, check that the environment variable `SIEBEL_LOG_EVENTS` is set to zero. To check, enter the following command at the MS DOS prompt: `echo %SIEBEL_LOG_EVENTS%` If this variable is not set, no action is required. However, if `SIEBEL_LOG_EVENTS` is returned with a value other than zero, you must set it to zero by performing the following steps:
 - Close Siebel Tools and any other Siebel client applications.
 - Navigate to Start > Settings > Control Panel > System > Environment.
 - In the Environment dialog box, in the System Variables box, select `SIEBEL_LOG_EVENTS`. Enter 0 in the Value box, and click Set. Click OK.
 - Relaunch Siebel Tools. The new setting becomes active.

NOTE: The steps you need to take to set this variable may vary depending on the operating system you are using.

2 Optimize your database, since database performance can cause the merge to slow down considerably.

- Make sure the database has sufficient rollback segments.

NOTE: If your RDBMS is Oracle, verify that your rollback segments are appropriately sized so that the largest of transactions can be accommodated.

- Make sure the database has enough space allocated.
- Make sure that the top-most logging applet in tools has no extra rows (from previous merge runs) when starting the merge.
- Make sure that the database is not loaded with users when merge is run (no other users should be connected).
- Delete extra repositories from the database using Siebel Tools. Running the merge on a database with more than the four (4) repositories that are needed for the merge will degrade merge performance. Before deleting extra repositories, make backups.

NOTE: Deletion of extra repositories may take a few hours.

- 3 Run the merge on a different machine than the database server. When you run the merge on a different machine, you can increase the foreground performance of the merge. To do this, increase the application performance setting on the machine and check that the Siebel Tools application is the foreground application.

To increase the foreground performance of the merge

- 1 Navigate to Start > Control Panel > System.
- 2 Select the Advanced tab.
- 3 Select the Performance Options button.
- 4 In the Application Response box, click the Applications radio button and click OK.
- 5 While the merge process is running, click on the title bar of the Siebel Tools application to ensure that the Siebel Tools application is the foreground application on the machine.

NOTE: After the merge process has finished, set the Performance setting back to its former value.

Performing the Repository Merge

The configuration utility that you ran in the section [“Installing and Configuring the Siebel Database Server Software” on page 109](#) loaded two version 7.5 standard repositories. You will now use Siebel Tools to merge your existing custom configuration into one of these new repositories, creating a version 7.5 custom configuration that includes all of your previous configuration changes.

The four repositories that currently exist in your development database are listed in [Table 15](#).

Table 15. Development Database Repositories

Repository Name	Description
Prior version 5.x, 6.x, or 7.0.x Siebel Repository	Standard version 5.x, 6.x, or 7.0.x repository, depending on the version from which you are upgrading.
Prior Customer Repository	Customized version 5.x, 6.x, or 7.0.x repository, depending on the version from which you are upgrading.
New Siebel Repository	Newly loaded version 7.5 standard repository.
New Customer Repository	Newly loaded version 7.5 repository into which your custom configuration will be merged.

Follow the guidelines provided in [“Improving Performance of the Merge” on page 144](#) to optimize performance of the repository merge.

- The merge is a memory-intensive process and will fail if insufficient memory is available on the Siebel Tools workstation. Before beginning a repository merge, make sure that the following preparations have been completed on the developer workstation. Make sure that the developer workstation on which Siebel Tools is running has been upgraded to the newest available version.
- Use a workstation with a minimum of 512 megabytes (MB) of RAM and at least 2-GB of virtual memory and a 2-GB page file. Otherwise, the merge will not complete successfully.
- Close all other applications before performing a repository merge.

NOTE: Some repositories may require additional memory, based on level of customizations.

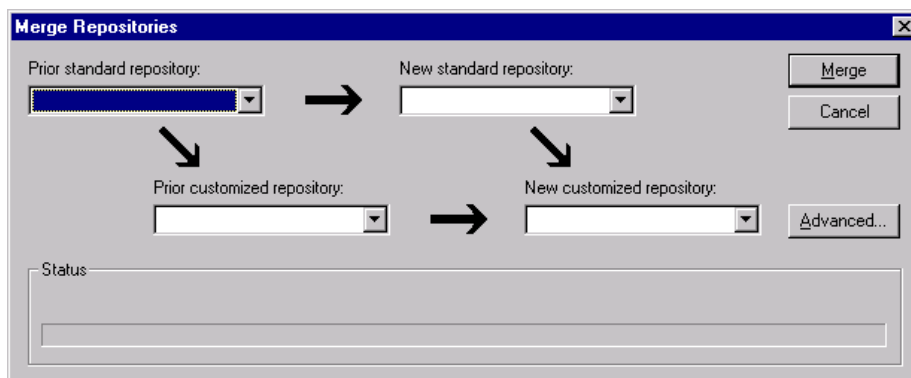
- If necessary, increase the swap space, using the Control Panel System applet, and then restart the development workstation before proceeding.

To merge the repository

- 1 Log on to a Siebel Tools client and use the File > Open Repository command to open the Prior Customer Repository.
- 2 Choose Tools > Upgrade > Upgrade Application.

CAUTION: Be sure to open the Prior Customer Repository, not another repository. Later steps in the repository merge process will fail if you have the wrong repository open.

The Merge Repositories dialog box appears.



- 3 In the Merge Repositories dialog box, choose the appropriate repository name from each picklist, using the repository names listed in the following table.

Drop–Down List Item	Value to Choose
Prior Standard Repository	Prior Release 5.x Siebel Repository, Prior Release 6.x Siebel Repository or Prior Release 7.0.x Siebel Repository, as appropriate for the version from which you are upgrading
Prior Customized Repository	Prior Customer Repository
New Standard Repository	New Siebel Repository
New Customized Repository	New Customer Repository

- 4 Review the settings in the Merge Repositories dialog box, then click Merge.

NOTE: The repository merge process may take, on average, five to seven hours to complete. Timings may vary greatly depending on the kind of machine, the hardware configuration, virtual memory allocation, the use of the upgrade inheritance feature, and level of customizations in the customer repository (such as new records or changed attributes). In addition to merging the base repository, all locales are merged. Additional time should be planned for each language, including the base language.

Customizations are moved to the New Customer Repository, which results in a large number of database operations (inserts and updates). For each of these operations, logging records are created, and these log records also affect performance. If the repository is large, or the database setup is not optimal, this may take much longer.

Reviewing the Merge Results

This section provides an overview of the repository merge process and describes how to review and modify, if desired, the results of the repository merge.

During the merge, errors are reported in the status view and recorded in the merge.txt file. There are no acceptable errors for the repository merge. The only exception to this rule is errors that occur as a result of the upgrade ancestor feature. These errors are acceptable. For more information, see [“Automatic Upgrade of Copied Objects” on page 127](#). After the merge has completed, you must review the results and resolve any errors and undesired conflicts before you proceed with the remainder of the upgrade.

NOTE: If an object that does not exist in the 7.x New Siebel repository is specified as an ancestor object, you will receive error messages in the merge.txt file. These errors are acceptable. An example of an error of this type is: `!!ERROR::CANNOT upgrade objects which have Briefing Tracking Profile Applet - Product marked as 'Upgrade Anc'`

Review errors through the Upgrade Applications Objects List screen. To get to the Upgrade Applications Objects List screen, from the Screens menu in Siebel Tools, choose Application Upgrader > Upgrade Application Objects List.

To view all errors encountered during the merge, query for `ERROR: :` in the Status field of the Object Differences applet (second applet on this view). Typically, all error records will be listed at the top.

Reviewing the merge.txt file to Determine the Cause of Errors

If errors are located, you need to review the merge.txt file. The merge.txt file is located in the `Tools\bin` directory. This file contains information that is logged while the merge is in progress. As errors are encountered during the merge, they are logged in the merge.txt file, preceded by the text `!!ERROR`.

NOTE: Each time you run the merge process, the merge.txt file is incremented and renamed with numbers appended to it. For this reason, you may find that your file has a name such as merge1.txt.

To search for logged errors in this file, search for exclamation points (!) throughout the file.

CAUTION: The merge.txt file can be used to figure out the cause of the error. In case of performance issues, this file can help you to determine where the time is being taken. Do not proceed if you encounter any errors as a result of the merge. See [“Recovering from a Failed Merge” on page 155](#). If you have any questions or require assistance in resolving these errors, please contact Technical Support or Professional Services.

Reviewing Merge Process Outcome

To review the overall outcome of the merge process

- 1 Navigate to the Application Upgrades object list view by choosing **Tools > Upgrade > Upgrade Application**. (You can also access the Application Upgrades object list view from the Screens menu by choosing **Screens > Application Upgrade**.)

The Application Upgrades applet at the top of this view lists all of the merge processes performed by the current user.

The Status field for a successful merge should read “Complete.”

- 2 If an error occurs, your merge has failed and you need to perform procedures to recover from a failed merge. To recover from a failed merge, see [“Recovering from a Failed Merge” on page 155](#).

Reviewing Object Differences

To review object differences

The Applications Upgrade object list view should still be displayed from the previous step. If the Applications Upgrade object list view is not displayed, navigate to it by choosing Tools > Upgrade > Upgrade Application.

- 1 From the Applications Upgrade object list view, click the Object Differences applet.

The Object Differences list displays new or deleted objects, or objects that had altered attributes. By default, the most important object differences are shown first, with errors appearing at the top of the list.

- 2 For each record, review the status field for errors. (You must tab to the right of the applet to see the status field.)
- 3 Query the applet for objects that do not have a check mark in the Attribute column; these are the new or deleted objects.

The various In... columns show which repositories the object existed in before the repository merge. The Add to New Customer Repository column indicates whether the object was merged into the New Customer Repository.

There are two types of object differences that must be resolved:

- Objects you previously deleted
- Objects deleted by Siebel eBusiness Applications

Customer-deleted objects. Any object that has a check mark in the “In Prior x.x Siebel Repository” and “Added to New Customer Repository” columns, but not in “In Prior Customer Repository,” was deleted during your customization of the Prior Customer Repository and automatically added back to the New Customer Repository. Typically, adding these objects back into your new repository does not cause a problem with your upgraded configuration. However, you need to examine the use of these objects carefully to make sure that this is the case.

Siebel-deleted objects. Any object that has a check mark in the “In Prior x.x Siebel Repository” and “In Prior Customer Repository” columns, but not in the “In New Siebel Repository” or “Added to New Customer Repository” columns, is obsolete in Release 7.5. Typically, when a new version of a similar object is introduced as part of a new release of Siebel eBusiness Applications, you must modify your upgraded configuration to make use of the new object.

The fields in the Object Differences list indicate common types of object definition differences. [Table 16](#) lists four common differences.

Table 16. Common Object Definition Differences from the Object Differences List

Common Object Type Difference	Conflict	Add to New Customized	In Prior Standard	In Prior Customized	In New Standard
Indicates an object definition that became obsolete in Release 7.5.			✓	✓	
Indicates an object definition added by the customer.		✓		✓	
Indicates an object definition that has been modified. The Attribute Differences list will then show the property differences.	✓	✓	✓	✓	✓
Indicates a new Release 7.5 object definition. This difference will not normally be shown. To show these object definitions, the Skip Logging... check box on the Advanced Merge Options dialog box must be turned off.					✓

Conflict Resolution

Reviewing Object Attribute Differences

Object attribute differences are shown in the Attribute Differences applets of the Application Upgrades Object List and Application Upgrades Attributes List views. The Object List view shows the differences for the currently selected object; the Attributes List view shows differences for all objects. You need to review any attribute differences that have a check mark in the Conflict column.

In either view, query all entries that have a check mark in the Conflict column, and ignore attributes of the following types:

- Left ■ Right ■ Top
- Height ■ Width

These are screen layout attributes that can be detected when running the Siebel client and corrected after you have completed the upgrade process.

The Resolution column shows whether the Standard Value (shown in the “In Prior x.x Siebel Repository” column) or the Custom Value (shown in the “Prior Customer Repository” column) was used as the object attribute in the New Customer Repository.

You can change the resolution for any conflict by entering a check mark for the Override flag. This automatically sets the attribute in the New Customer Repository to the opposite value.

CAUTION: Do not rerun the repository merge into the New Customer Repository after entering the Override flag check mark. If you run the merge again, you will remove the Override attribute.

You can override attribute differences only as long as each repository used during the merge process remains in its current state in the database server. Resolve all attribute conflicts before continuing with the upgrade process.

Examples of Conflict Resolution

The following examples demonstrate how conflicts are resolved during the merge.

A customer change that would be preserved by the merge. If the value in Prior Customer Repository is unique, and the values in Prior Siebel Repository and New Siebel Repository are the same, the customer change will be preserved in the New Customer Repository.

For example, if the following statements are true,

- Prior Siebel Repository has value 50 for the property `height` of applet `x`
- Prior Customer Repository has value 25 for the property `height` of applet `x`
- New Siebel Repository has value 50 for the property `height` of applet `x`

then the value of 25 will be set in the New Customer Repository.

A customer change that would be overridden by the merge. If the values in all three repositories are different, then the default value from the previous release was changed by the customer, but also changed in the new release. The two updated values result in a conflict. In 90% of cases where there is a conflict, the value from New Siebel Repository will override the value in the Prior Customer Repository.

For example, if the following statements are true,

- Prior Siebel Repository has value 20 for the property `height` of applet `x`
- Prior Customer Repository has value 25 for the property `height` of applet `x`
- New Siebel Repository has value 50 for the property `height` of applet `x`

then the value from the New Siebel Repository will be merged into the New Customer Repository.

Certain attributes determine the 10% of cases where the value from the Prior Customer Repository will override the value in the New Siebel Repository. View the attribute under object type `TYPE` in Tools.

If the `StandardWins` flag is set to `Y`, the New Siebel Repository value is chosen.

CAUTION: If the `StandardWins` flag is set to `N`, then the Prior Customer Repository value is chosen. The default `StandardWins` settings delivered with your Siebel eBusiness application are highly recommended. Avoid changing the `StandardWins` settings.

Troubleshooting

Recovering from a Failed Merge

If the repository merge process fails for any reason (for example, due to a lack of virtual memory on the Siebel Tools workstation, or a database error), you must perform the following steps to restore your pre-merge database:

- Fix the problem that caused the merge to fail.
- Restore your database from the backup that you took after you upgraded the Siebel database schema (upgrep). See [“Taking a Backup of the Database Repository” on page 123](#).
- Prepare the prior customer repository for the merge. See [“Preparing the Prior Customer Repository for the Merge” on page 124](#).
- Rerun the merge. See [“Performing a Repository Merge, Using Siebel Tools” on page 143](#).

Generating EIM Temporary Columns

The repository merge process does not preserve EIM processing columns for custom mappings. After the completion of a successful merge, you need to execute the steps below to generate the missing custom EIM processing columns.

To generate EIM temporary columns

- 1** Open Tools on “New Customer Repository.”
- 2** Go to the Tools menu > Upgrade > Generate EIM Processing Columns.
- 3** At the dialogue box, click OK to generate EIM processing columns for custom mappings.
- 4** Verify that this process has completed successfully by reviewing the Comment field on the new repository within Siebel Tools. The Comment field should show UpgEimCol.

NOTE: The repository that you run this process on must be the one that you use to upgrade the Custom Database Schema (upgphys).

Updating User Responsibilities

You need to update user responsibilities that may have been created in the Siebel MidMarket Edition applications Release 5.x. Note that data changes such as creating or changing User Responsibilities do not need to be migrated to the user community. This data will be accessible when the user next logs into the Siebel MidMarket Edition application Release 7.x and connects to the updated production database.

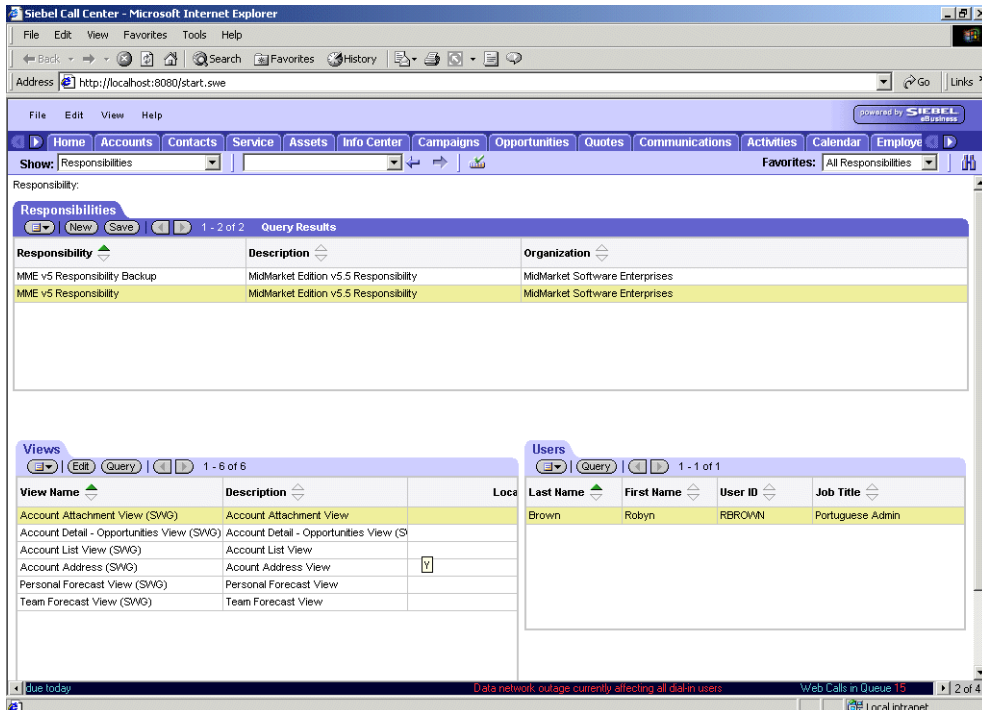
For an explanation of User Responsibilities, see *Applications Administration Guide, MidMarket Edition*.

NOTE: If custom User Responsibilities were not created in the Siebel MidMarket Edition Applications Release 5.x, then skip this section. Your repository upgrade is complete.

To update user responsibilities

- 1 Launch your employee application, such as Call Center or Sales Release 7.x, and log in as the Administrator.

- 2 Go to View > Site Map > Application Administration > Responsibilities. Select one of the custom Responsibilities that was created in the Release 5.x applications. In the figure below, the example Responsibility is called “MME v5 Responsibility.”



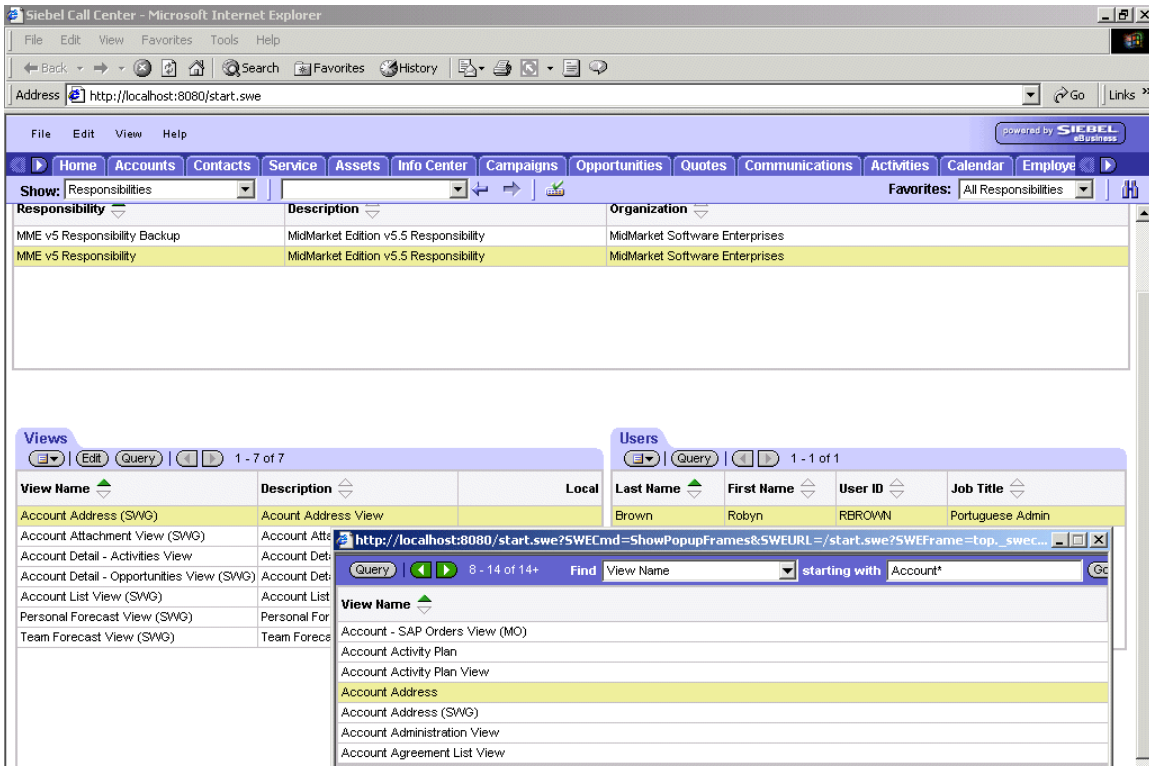
Robyn Brown is the only user assigned to the Responsibility “MME v5 Responsibility.” Also note that there are only a handful of views associated to this responsibility.

- 3 Since the “(SWG)” objects were merged into the Siebel 7.x repository, none of the objects have suffix “(SWG)” like they did in Release 5.x. To update this responsibility, add views that have the same names as the views in the Responsibility without the “(SWG)” suffix. Then, delete the views with the “(SWG)” suffix. The following steps will lead you through this process.
- 4 Before proceeding with this task, make a backup of the responsibility from the Siebel MidMarket Edition v5.x application simply by copying and renaming it. “MME v5 Responsibility Backup” is the backup in this example.

Upgrading the Development Environment

Performing a Repository Merge, Using Siebel Tools

- 5 The first view that needs to be added in this example is “Account Address.” Select the menu button on the View applet, and select New Record. In the popup window, select “View Name” for Find, and enter “Account*” for starting with. Click “Go” to find all records starting with “Account.” Select “Account Address” and click OK. See the figure below to add this view.



- 6 Next, delete the corresponding view with the “(SWG)” suffix. The first view that will be deleted in this example is “Account Address (SWG).”

- 7** Repeat steps 3 through 6 until all views in the custom-built responsibility from the Siebel MidMarket Edition v5.x are renamed.

CAUTION: There are two things that you must be aware of when updating the user responsibilities:

Do not rename any views that were completely custom-built in the Siebel v5.x applications. These are new views to the Siebel repository and do not need to be renamed since they were not merged with anything in the new Siebel 7.x repository.

Do not rename the “Personal Forecast View (SWG)” and “Team Forecast View (SWG)” views. If you are not using these views, please disregard this note. If you are using these views, again, do not rename them. The “(SWG)” suffix must remain a part of the view names. The figure below shows the “MME v5 Responsibility” after all relevant v5.x view names have been updated to Siebel MidMarket Edition v7.x names.

Upgrading the Development Environment

Performing a Repository Merge, Using Siebel Tools

Figure 6 displays the Responsibilities view after all MidMarket version 5.x Responsibility view names have been updated.

Siebel Call Center - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Back Forward Stop Reload Search Favorites History Print View Source

Address http://localhost:8080/start.swe

File Edit View Help

Home Accounts Contacts Service Assets Info Center Campaigns Opportunities Quotes Communications Activities Ca

Show: Responsibilities

Favorites:

Responsibilities

New Save 1 - 2 of 2 Query Results

Responsibility	Description	Organization
MME v5 Responsibility Backup	MidMarket Edition v5.5 Responsibility	MidMarket Software Enterprises
MME v5 Responsibility	MidMarket Edition v5.5 Responsibility	MidMarket Software Enterprises

Views

Edit Query 1 - 7 of 7

View Name	Description	Local
Account Detail - Opportunities View	Account Detail - Opportunities View	
Account Attachment View	Account Attachment View	
Account Address	Account Addresses (Bill To / Ship To)	
Account Detail - Activities View	Account Detail - Activities View	
Account List View	Account Listing	
Personal Forecast View (SWG)	Personal Forecast View	
Team Forecast View (SWG)	Team Forecast View	

Users

Query 1 - 1 of 1

Last Name	First Name	User ID	Jo
Brown	Robyn	RBROWN	Po

401k enrollment forms due today

Data network outage currently affecting all dial-in users

Web Calls in Queue 15

Figure 6. Responsibilities View

The next time that Robyn Brown logs into a Siebel application, he will have access to the new versions of all views to which he is accustomed to having access. These views will contain the customizations made to them in v5.x as well as the new functionality available in the Release 7.x product.

Upgrading the Custom Database Schema

The configuration utility that you ran in [“Installing and Configuring the Siebel Database Server Software” on page 109](#) updated the Siebel Database Schema (upgrep). You will now use the Siebel Software Configuration Utility to apply these changes to the custom database schema (upgphys) to upgrade it to the new version.

The repository on which you run the Custom Database Schema upgrade must be the same repository that you prepared for the merge in [“Generating EIM Temporary Columns” on page 155](#).

To upgrade the custom database schema (upgphys)

- 1 Verify that no server tasks are running in the background.

If necessary, stop Siebel Servers and Siebel Gateway Name Server service by navigating to Start > Settings > Control Panel > Services.

- 2 Launch the Siebel Software Configuration Utility by selecting Start > Programs > Siebel Enterprise Servers 7.0 > Configure DB Server.

The Siebel Enterprise Parameters: Gateway Server Address screen appears.

- 3 Type the following values as you recorded them in your copy of [“Upgrade Planning Worksheet”](#):

- **Gateway Server Address.** The alias of your Siebel Gateway Name Server (typically the machine name).
- **Enterprise Server Name.** The name of your Enterprise Server, for example, siebel.

To continue, click Next.

The Installation and Configuration Parameters: Siebel Server Directory screen appears.

- 4 Accept the default value displayed in the Siebel Server Directory field, or type an alternate directory path for your configuration. Click Next.

NOTE: This is the *SIEBSVR_ROOT* directory, for example,
D:\sea7xx\siebsvr.

The Installation and Configuration Parameters: Siebel Database Server Directory screen appears.

- 5 In the Siebel Database Server Directory field, verify that the default directory path displayed is the correct database server installation directory for your configuration. If it is not, use the Browse button to navigate to a different database directory.

NOTE: This is the *DBSRVR_ROOT* directory, for example, D:\sea7xx\dsrivr.

To continue, click Next.

- 6 In the Siebel Database Server Options : RDBMS Platform screen, choose the database platform that you are upgrading.

- **IBM DB2 for Unix and Windows.**
- **Microsoft SQL Server.**
- **Oracle.**

The Siebel Database Server Options: Siebel Database Operation menu appears.

- 7 Choose `Upgrade Database` and click Next.

The Upgrade the Siebel Database: Upgrade Options screen appears.

- 8 Choose `Upgrade Custom Database Schema (upgphys)` and click Next.

The Upgrade Configuration Information: Environment Type screen appears.

- 9 Choose `Development` to upgrade your development environment.

To continue, click Next.

The Upgrade Configuration Information: Current Siebel Version screen appears.

- 10** Choose the base version of Siebel eBusiness Applications from which you are upgrading.

To continue, click Next.

The Installation and Configuration Parameters: Language Selection screen appears.

NOTE: The language selection screen will not appear if you have only one language installed, because that language is automatically defined as the primary language. Therefore, you will skip the next step and continue with [Step 12](#).

- 11** Choose the language that was the primary language of your prior environment.

To continue, click Next.

NOTE: If you cannot continue, then you selected a language for which you do not have a language pack. You need to reselect the primary language used in your prior environment, and then click Next to continue.

The next Installation and Configuration Parameters: ODBC Data Source Name screen appears.

- 12** Accept the name of the ODBC data source displayed for verification purposes (for example, `SiebSrvr_siebel`), or enter a different data source name.

NOTE: The data source is created automatically by the Siebel Server installation, using the format `SiebSrvr_EnterpriseName`.

To find the name of your ODBC data source, navigate to Start > Settings > Control Panel > Administrative Tools > Data Source (ODBC). Click the System DSN tab and you will find the name of your ODBC data source.

To continue, click Next.

The Installation and Configuration Parameters: Database User Name screen appears.

13 Type the source user name and password for your database:

- **Database User Name.** User name of the Siebel administrator, for example, `sadmin`.
- **Database Password.** Password for the Siebel administrator, for example, `sadmin`.
- **Database Password (confirm).** Retype the password for confirmation.

To continue, click Next.

NOTE: The following Installation and Configuration Parameters screens are platform-specific. Which screens appear next will depend on the database platform that you are upgrading.

- 14** In the Installation and Configuration Parameters screens that appear next, complete each field with the values that you recorded on your Upgrade Planning Worksheet. The default values for each RDBMS are listed in the table below.

NOTE: Use underscores rather than spaces; these values are case-sensitive.

Field	Value
Database Table Owner	<p>This is the account that will own the Siebel objects, for example, <code>siebel</code>.</p> <ul style="list-style-type: none"> ■ DB2 UDB: Tableowner. ■ Microsoft SQL Server: Database Owner Login (this is the login for the owner of the database, not necessarily the default owner of the database in DBO). ■ Oracle: Tableowner.
Database Table Owner Password	<ul style="list-style-type: none"> ■ DB2 UDB: Tableowner password. ■ Microsoft SQL Server: the password for Database Owner Login (this is the login for the owner of the database, not necessarily the default owner of the database in DBO). ■ Oracle: Tableowner password.
Index Space	<ul style="list-style-type: none"> ■ DB2 UDB: The name you give to your 4-K index space for tables. ■ Oracle: The name you gave to your index area.
4KB Table Space	<ul style="list-style-type: none"> ■ DB2 UDB: The name you gave to your 4-KB table space.
8KB Table Space (DB2 UDB upgrades from 5.x only)	<ul style="list-style-type: none"> ■ DB2 UDB: The name you gave to your 8-KB table space.
16KB Table Space (DB2 UDB only)	<ul style="list-style-type: none"> ■ DB2 UDB: 16 KB Table space.
32KB Table Space (DB2 UDB only)	<ul style="list-style-type: none"> ■ DB2 UDB: 32 KB Table space.
Table Space Name	<ul style="list-style-type: none"> ■ Oracle: The name you gave to your data area.

After you type the value for each screen, to continue, click Next.

- 15** For all platforms, in the Upgrade Configuration Information: Database Server OS screen, choose the platform on which your database server runs; for example, Windows.

To continue, click Next.

The Configuration Parameter Review screen appears.

- 16** Review the configuration values you entered on the previous Configuration Utility screens against the values that you recorded in your copy of [Appendix A, “Upgrade Planning Worksheet.”](#)

NOTE: Passwords are encrypted and will not appear in plain text either in the user interface or in the upgrade configuration files (UCF files). After a password is entered, it will always appear in encrypted form. If you need to use another password, you must re-run the configuration utility.

- If you need to go back to make changes, click Previous to back out until you reach the screen with the parameter you need to change. Enter the valid parameter, and then click Next until you reach the Configuration Parameter Review screen again.
 - To accept the values you input with no changes, click Finish.
- 17** A message box appears, prompting you to decide if you want to apply the configuration now or later.
- To apply the configuration now, click OK.

The Siebel Upgrade Wizard appears. To begin the upgrade of your Custom Database Schema, click OK. A check mark will appear beside each item as it is completed. When the status bar registers that the upgrade process is complete, click OK to exit the Siebel Upgrade Wizard.

At this stage, you have finished upgrading your Custom Database Schema and are ready to review the log files for errors.

- To apply the configuration later, click Cancel.

When you are ready to apply your configuration, follow the procedure [“To apply your Siebel Software configuration later” on page 117](#), [“To apply your Siebel Software configuration later.”](#)

CAUTION: Some long-running operations, when canceled, may not sever all processes on the database side. If your upgrade does not complete successfully, work with your database administrator to verify that there are no other users currently logged on to the database. If there are, ask your database administrator to terminate all connections, then relaunch the configuration wizard.

If errors are encountered during the upgrade process, it will stop at that point. If your upgrade stops due to an error, you must carefully review the log files to make sure that your upgrade has completed successfully up to that point, and fix the error that stopped the upgrade. See [“Troubleshooting the Custom Database Schema Upgrade” on page 168](#). Once you have corrected the error, you may restart the upgrade, and it will continue from the last step that completed successfully. For details on how to restart, see [“Restarting the Upgrade” on page 168](#).

To apply your Siebel Software configuration later

- 1 If you want to apply the configuration later, you must first review the upgrade configuration file (UCF file) under the *SIEBEL_ROOT\siebsrvr\bin* directory:

```
master_<upgrade_option>_<upgrade_type>_<version>.ucf
```

for example,

```
master_upgphys_dev_601.ucf
```

- 2 After you have reviewed the UCF file, to apply the configuration, enter the following directory path in the DOS command prompt:

```
SIEBEL_ROOT\siebsrvr\bin\siebug.exe /m master_<upgrade_option>_<upgrade_type>_<version>.ucf
```

for example,

```
SIEBEL_ROOT\siebsrvr\bin\siebug.exe /m  
master_upgphys_dev_601.ucf
```

Troubleshooting the Custom Database Schema Upgrade

Typical problems that might occur during the upgrade of your custom database schema (upgphys) could result from a lack of storage space or insufficient user privileges.

Reviewing the Upgrade Log Files

The upgrade wizard creates several log files, such as UpgWiz.log, UpgWiz_01.log (the name of the log file will increment for subsequent log files) within the `SIEBEL_ROOT\siebsrvr\LOG` subdirectory. These log files may include errors that are expected and benign: acceptable errors are documented in the `errors.rtf` file located in the `siebsrvr_root\log` directory. You must review the log files carefully for unacceptable errors.

To review the log files, see [“Reviewing the Upgrade Log Files” on page 119](#).

CAUTION: Do not proceed with the upgrade until unacceptable errors have been corrected. If you cannot correct the error, contact Siebel Technical Support or Professional Services to report the error in detail.

Restarting the Upgrade

The Siebel Upgrade Wizard is restartable at most stages within the upgrade process. If the Siebel Upgrade Wizard encounters an error during the upgrade process, it will stop at that point. If your upgrade stops due to an error, you must carefully review the log files to make sure that your upgrade has completed successfully up to that point. Once you have verified this and resolved the failure, you may restart the upgrade. The upgrade will continue from the last step that completed successfully.

CAUTION: Before you restart the upgrade (after any break in the upgrade process) back up your complete set of log files, from the beginning of the process to the point at which it stopped, to another directory. This will maintain a complete record of your log files, and prevent your previous log files from being overwritten, which might prevent accurate diagnosis of the reason for the break in the upgrade.

To manually launch the Siebel Upgrade Wizard from the command prompt, or to restart the upgrade

- 1 Type the following command at the Windows DOS command prompt from the `SIEBEL_ROOT\siebsrvr_root\BIN` directory:

```
siebug.exe /m master_<upgrade_option>_<upgrade_type>_<version>.ucf
```

For example, to restart a development upgrade from Release 6.0.1 to Release 7.x, type the following command:

```
siebug.exe /m master_upgphys_dev_601.ucf.
```

The Siebel Upgrade Wizard screen appears, displaying the items to be completed during this phase.

- 2 To begin the upgrade of your custom schema, click OK.
A check mark will appear beside each item as it is completed.
- 3 When the status bar registers that the upgrade process is complete, click OK to exit the Siebel Upgrade Wizard.

You have finished upgrading your repository and are ready to review the log files for errors. See [“Reviewing the Upgrade Log Files” on page 168](#).

Migrating Custom Business Component Configurations

Customized business component configurations (buscomps) which are based on customer extension columns or on Siebel columns in obsolete tables need to be identified and manually fixed after the upgrade. Siebel columns in obsolete tables are reconfigured and migrated during the upgrade. However, customer buscomps or custom extension columns on obsolete tables need to be migrated manually.

NOTE: Custom extension columns on tables that are upgraded (not obsolete) are retained during the upgrade.

To review a list of the buscomps that require manual migration, review the `upgcust.log` file, located in the `SIEBSRVr/log` directory.

Critical obsolete tables are listed in [Table 17](#).

Table 17. Repository Tables That Are Obsolete in Release 7.x

Previous Table	Suggested New Table
S_EMPLOYEE	S_CONTACT, S_USER, S_EMP_PER
S_EMP_POSTN	S_PARTY_PER
S_ORG_INT	S_ORG_EXT, S_BU
S_POSTN_RPT_REL	S_PARTY_RPT_REL

The access control buscomp migration utility is run on the Prior Customer Repository by the Upgrade Wizard during upgrade of the Siebel database schema.

Several fields, buscomps, and columns may need to be reevaluated and recreated after your upgrade. A sample conversion script will be available for migration of data to the new columns during the development upgrade.

CAUTION: Do not proceed further with the upgrade until such errors have been analyzed and the necessary corrective action taken.

Development Environment Postupgrade Tasks

5

This chapter describes the tasks which you need to perform after the upgrade of the Siebel eBusiness Applications development environment.

After you have successfully completed the upgrade steps in [Chapter 4, “Upgrading the Development Environment,”](#) continue with the post-upgrade tasks listed in [Table 18](#).

Table 18. Siebel Development Environment Postupgrade Tasks

Postupgrade Tasks	
1	Drop DB2 8-KB table spaces and buffer pools. See “Dropping DB2 8-KB Table Spaces and Buffer Pools” on page 173.
2	Manually archive the log files. See “Manually Archiving the Log Files” on page 173.
3	Reapply custom extensions on obsolete tables. See “Reapplying Custom Extensions on Obsolete Tables” on page 177.
4	Update file system directory. See “Updating the File System Directory” on page 174.
5	Update file system attachments. See “Updating File System Attachments” on page 175.
6	Resolve business component and join conflicts. See “Resolving Business Component and Join Conflicts” on page 180.
7	Identify Obsolete Objects. (Optional) See “Identifying Obsolete Objects” on page 181.
8	Set visibility modes for access control. See “Setting Visibility Modes for Access Control” on page 183.
9	Resolve duplicate EIM mappings. See “Resolving Duplicate EIM Mappings” on page 186.
10	Safeguard the Custom Repository export file. See “Safeguarding the New Custom Repository Export File” on page 187.
11	Produce a new custom configuration file. See “Producing a New Custom Configuration File” on page 188.
12	Regenerate the database template file. See “Regenerating the Database Template File” on page 189.
13	Extract Developers or Siebel Tools Clients. See “Extracting Developers or Siebel Tools Clients” on page 189
14	Test the Upgraded Configuration. See “Testing the Upgraded Configuration” on page 189.
15	Validate the upgrade. See “Validating the Upgrade” on page 190.
16	Run statistics. See “Run Statistics” on page 191.
17	Reorganize the P1 index for DB2 UDB. See “Reorganize the P1 Index for DB2 UDB” on page 191.
18	Reset database server configuration parameters. See “Reset Database Server Configuration Parameters” on page 191.
19	Review additional postupgrade tasks and considerations for specific applications. See “Additional Postupgrade Tasks for Specific Applications” on page 192.
20	Upgrade your encryption method to RC2 after your upgrade is complete and before you deploy your application. See “Upgrading Your Encryption Method” on page 199.
21	Set up your environment for support global time zone. See “Setting Up Your Environment to Support Global Time Zone” on page 202.
22	Review planning considerations for migration from a non-Unicode code page to Unicode. See “Migrating to Unicode” on page 205.

Postupgrade Tasks for All RDBMS Development Environments

This section describes postupgrade tasks for RDBMS development environments.

Dropping DB2 8-KB Table Spaces and Buffer Pools

If you are upgrading a database under Windows, you should now drop your 8-KB table space, 8-KB temporary table space, and 8-KB buffer pool. Before dropping your 8-KB table space, check for the existence of any tables in it by running the following SQL statement:

```
select name from sysibm.systables where TBSPACE='TBS_8K'
```

CAUTION: Check your 8-KB table space to make sure that all objects have been migrated to your new 16-KB table space before dropping the 8-KB table space. Otherwise, you will lose this data.

Manually Archiving the Log Files

After a successful installation and upgrade, you must save and archive the log files located in the `siebsrvr_root/log` directory. This is a manual process.

By default, only nine (9) log files are retained for subsequent retries of the upgrade wizard. After nine log files have been created, when the upgrade wizard is rerun, it will overwrite log files beginning with the earliest one created and recycle the rest as necessary.

The number of log files retained can be increased by resetting the `siebel_log_archive` environment variable to 20, for example, to retain twenty (20) log files.

Updating the File System Directory

In your previous installation, the file system had a flat structure with one directory. The Release 7.x file system contains subdirectories. During the upgrade of your production environment, Siebel Anywhere will look for certain files in the file system subdirectories, but these files only exist in the root file system directory.

Therefore, you need to create the appropriate subdirectories in your previous installation directory structure and then copy the required files from the root file system directory to the new subdirectories.

To make file attachments accessible by Siebel eBusiness Applications

- 1 In the existing file system structure, create an `\att` subdirectory if it does not already exist; for example, `\\siebfile\att`

where:

siebfile is your root file system directory.

NOTE: Creating the `\att` subdirectory will not adversely affect the installation of your Siebel Server. (You will install the Siebel Server at a later point.)

- 2 Copy all files located under the `\\siebfile` directory to the `\\siebfile\att` directory so that all file attachments will be accessible by Siebel eBusiness Applications.
- 3 Verify that files have copied correctly to the `\\siebfile\att` directory. After this has been verified, clean up the file system.

CAUTION: On the Windows platform, client installation directory paths are limited to 18 characters. If your previous installation directory name was more than 18 characters (for example, `C:\YOURCOMPANYNAME\siebel`, where *YOURCOMPANYNAME* is more than 12 characters long), you need to choose a new installation directory. Support for specifying a new installation directory is provided by SiebelAnywhere.

Updating File System Attachments

During the Siebel database upgrade, data from the inactive table `S_EMPLOYEE` was migrated to `S_CONTACT`, `S_USER`, `S_EMP_PER` and data from `S_ORG_INT` was migrated to `S_ORG_EXT`. Attachments in `S_ORG_INT_ATT` were migrated to `S_ACCNT_ATT` and attachments in `S_EMPLOYEE_ATT` were migrated to `S_CONTACT_ATT`.

NOTE: If your upgrade to Release 7.5 was from Release 7.0.3 or 7.0.4, files from `S_WEB_CNTNT` were migrated to `S_CB_ASSET_VER`.

Data migration affects the physical file structure of the Siebel File System. Therefore, you need to run a utility to rename file attachments which correspond to inactive tables for Release 7.x so that they will be accessible by Siebel eBusiness Applications. For example, this utility will copy and rename all files named `S_EMPLOYEE*.SAF` to `S_CONTACT*.SAF` and all files named `S_ORG_INT_ATT*.SAF` to `S_ACCNT*.SAF` so that they correspond to new table names.

To upgrade file attachments

- Run `chng_file_sys.bat`.

From `SIEBSRV_ROOT\bin`, enter the following command:

```
chng_file_sys.bat <Source_Table> <Target_Table>  
<"File_System">
```

where:

- `Source_Table` = name of the original, inactive, table
- `Target_Table` = name of the new table to which the original data was migrated
- `"File_System"` = name of the directory where the file system resides (entered inside quotation marks)

for example,

```
chng_file_sys.bat S_EMPLOYEE S_CONTACT  
"D:\DBSRVR_ROOT\Files"
```

Review the renamed files carefully to verify that they can be accessed by Siebel eBusiness Applications.

CAUTION: Make sure that attachment files for obsolete tables are renamed or copied to alternate locations.

For example, since `S_EMPLOYEE_ATT` is migrated to `S_CONTACT_ATT`, you need to rename a file such as `S_EMPLOYEE_12-1ABC.SAF` to `S_CONTACT_12-1ABC.SAF`.

If you upgraded from Release 7.0.3 or 7.0.4, data in `S_LIT` was migrated to `S_CB_ASSET_VER`; therefore, you need to migrate files associated with migrated records. Run the following utility to copy the files named `S_LIT*.SAF` to `S_CB_ASSET_VER*.SAF` so that the files correspond to the new table name.

To upgrade `S_LIT` files to `S_CB_ASSET_VER` files

- Run `file_upg_mm.bat`

From `SIEBSRV_ROOT\bin`, enter the following command:

```
file_upg_mm.bat Odbc_Source User_Name Password Table_Owner  
File_System_Location Siebel_Home_Directory  
Siebel_Db_Server_Home_Directory
```

where:

- `Odbc_Source` = the ODBC source of the database
- `User_Name` = the database user name
- `Password` = the password for the database user name
- `Table_Owner` = the database table owner
- `"File_System_Location"` = the directory where the file system resides (entered inside quotation marks)
- `"Siebel_Home_Directory"` = the directory where Siebel Server is installed (entered inside quotation marks)
- `"Siebel_Db_Server_Home_Directory"` = the directory where Siebel Database Server is installed (entered inside quotation marks)

for example,

```
file_upg_mm.bat SEBL sadmin sadminpw SIEBEL  
"D:\DBSRVR_ROOT\Files" "D:\SEBLSRVR_ROOT" "D:\DBSRVR_ROOT"
```

Review the renamed files carefully to verify that they can be accessed by Siebel eBusiness Applications.

Reapplying Custom Extensions on Obsolete Tables

Several tables used in prior versions of Siebel eBusiness Applications are no longer used, or have been replaced by new tables in Release 7.x. The new configuration uses the new tables. You need to reapply the extensions that you added to previous versions of the tables to the new tables.

The Siebel Software Configuration Utility, described in [“Upgrading the Custom Database Schema” on page 161](#), generates a report that you must review for information about tables that are either obsolete or no longer in use for Release 7.x. This report, `xtndobstbl.txt`, lists extension columns that reside on obsolete tables, and therefore need to be moved to alternate tables. Each table is listed with one of three status codes:

- **Not Used.** These tables are not used in Release 7.x, but you may continue to use them; for example, `S_ORG_PROD`, `S_PERIOD BU`. These tables are supported as is (for instance, with docking or EIM).
- **EOL (end of life).** These tables are not used in Release 7.x, and they will not be supported in future releases; for example, old `S_*_IF` tables.
- **Inactive.** These tables have been discontinued, and are not supported in Release 7.x. Extension columns that reside on inactive tables must be moved to alternate tables; for example, `S_EMPLOYEE` should be moved to `S_CONTACT`, `S_USER`, or `S_EMP_PER`.

If no tables are listed in `xtndobstbl.txt`, no action is required. If this file lists any tables, their custom extensions must be reapplied to Siebel eBusiness Applications Release 7.x tables using Siebel Tools. See *Siebel Tools Reference, MidMarket Edition*.

Table 19 lists examples of previously used tables which are inactive in Release 7.x (you can no longer use these tables) and lists the suggested new tables to which custom extensions need to be reapplied. The new tables are recommendations only; the tables that you need to apply the extensions to might vary depending on their type and use. Contact Siebel Technical Services to validate the reapplication of extension columns and to review the steps necessary to migrate any extension column data to the new tables.

This data will have to be migrated during both the development and production environment upgrades.

Table 19. Examples of Tables That Are Obsolete in Release 7.x

Previous Table	Suggested New Table
S_EMPLOYEE	S_CONTACT, S_USER, S_EMP_PER
S_EMP_POSTN	S_PARTY_PER
S_ORG_INT	S_ORG_EXT, S_BU
S_POSTN_RPT_REL	S_PARTY_RPT_REL

If you have created many custom extension columns on the tables `S_EMPLOYEE` or `S_ORG_INT`, both of which are no longer used in Release 7.x, the joins between the tables will not be accurate. This may result in SQL errors when you launch the Siebel client.

In such cases, using Siebel Tools, you need to manually create corresponding extension columns in the new target tables, and manually move the data to the new extension column on the new table before you continue migration of the application. You should then review the business component configuration to make sure that the client will operate properly.

You may need to do this in one of the following instances:

- Fields based on custom extension columns in `S_EMPLOYEE` or `S_ORG_EXT`
- Fields based on custom extension tables from `S_EMPLOYEE` or `S_ORG_INT` with or without join
- Custom joins to custom extension tables from `S_EMPLOYEE` or `S_ORG_INT`

If you review the `xtndobstbl.txt` file after you run the upgrade, you will find a list of fields that require your attention.

[Table 20](#) lists examples of previously used tables that are no longer used in Release 7.x, but that you may want to continue to use.

Table 20. Examples of Tables That Are Not Used in Release 7.x

Previous Table	Suggested New Table
S_CRSE	S_SRC, S_SRC_EVT
S_CRSE_OFFR	S_SRC, S_SRC_EVT
S_CRSE_REG	S_SRC_REG
S_CTLG_CAT_REL	S_CTLG_CAT
S_OPTY_PROD	S_REVN
S_TMSHT_LINE	S_TMSHT_ITEM, S_TMSHT_LN

[Table 21](#) lists examples of tables which were unused in previous releases of Siebel eBusiness Applications, but are now used in Release 7.x.

Table 21. Examples of Previously Unused Tables That Are Used in Release 7.x

Now Used Table	Used to be...
S_ACT_EMP	S_EVT_ACT
S_ACT_CON	S_EVT_ACT

Resolving Business Component and Join Conflicts

After the Upgrade Siebel Database Schema phase, you may need to perform manual steps to business component fields and joins, depending upon the complexity of your business component configuration. You should thoroughly review the post upgrade configuration to make sure that the object level definitions are preserved as expected. During the upgrade, a list of business component joins and fields that need to be manually rectified are generated in a log file titled, `upgcust.log`. This particular log file, along with others generated by the upgrade process, can be found under `SIEBEL_ROOT/log`.

The log file contains two distinct sections:

- Part 1

For extension columns on obsolete tables such as `S_EMPLOYEE` and `S_ORG_INT`, you need to reimplement the extension columns on the replacement tables. After you have done this, review the business component definition to verify proper operation.

The report generated by part 1 of the log file provides a list of the business component fields that are based on custom extension columns in obsolete tables such as `S_EMPLOYEE` and `S_ORG_INT`. This list displays the following properties:

- Business Component Name
- Field Name
- Column Name

The table `S_EMPLOYEE` is migrated to three tables, `S_CONTACT`, `S_EMP_PER` and `S_USER`. After you have determined and implemented the approach for previously defined custom extension columns on obsolete tables, you should manually configure the business component field to reference that database column. Any manual reconfiguration must be done in the New Customer Repository after the upgrade has been run.

■ Part 2

Because the data from S_EMPLOYEE and S_CONTACT has moved into more than one table, there is potential for conflicts between custom configuration from a previous release and standard configuration in 7.x. You need to resolve these conflicts in order for the application to function as designed.

After the repository merge has been run, there may be inconsistencies in the join names and joins set at the field level. If you do not resolve these discrepancies, it is likely that the application configuration will result in errors or will result in incorrect behavior. The report generated by part 2 of the log file provides a list of joins that were not updated during the merge process. This list displays the following properties:

- Business Component Name
- Join Name

With this list, you will need to go to each of the Business Component definitions and manually change the join name from the current value to the value listed in the report. Make sure that there is consistency between the joins as defined by name and the joins defined for each of the fields. Again, any manual configuration must be done in the New Customer Repository after the upgrade has been run.

Identifying Obsolete Objects

In Release 7.5, many objects have become obsolete. After performing the repository merge, you can generate a list of obsolete objects using Siebel Tools. Objects that were available in the Prior Standard repository are compared with the objects that are available in the New Standard repository. All objects that were available in the Prior Standard repository but are not available in the New Standard repository are obsolete in Release 7.5.

NOTE: This procedure is optional.

To generate a list of obsolete objects

- 1** On the Siebel Tools workstation on which you performed your successful repository merge, select Screens > Application Upgrader > Application Upgrade Object List.
- 2** In the Application Upgrade list, select the record of the successful merge.
- 3** In the Object Differences list, click Query.
- 4** Enter your query criteria in the Object Differences list:
 - Click in the In Prior Standard checkbox so that a checkmark appears in the checkbox.
 - Click in the In New Standard checkbox and then click the checkbox again so that a check does not appear in the checkbox.
 - Click in the In Prior Customized checkbox so that a checkmark appears in the checkbox.
 - Click in the Attribute checkbox and then click the checkbox again so that a check does not appear in the checkbox.
- 5** Press Enter to run the query.

All obsolete objects in Release 7.5 appear in the Object Differences list. You can filter the objects displayed by using the Top Parent Type and Object Type fields.

Setting Visibility Modes for Access Control

Before you deploy the upgraded configuration in a test or production environment, you need to make a decision about what type of visibility to use on certain business components, views, and applet picklists. This requires some evaluation. Certain areas of the Release 7.x default configuration use Catalog visibility.

- If you are prepared to define the catalog, category, and access group structure in parallel with the postupgrade development and testing phase before production rollout, then changes to visibility properties might not be necessary. This decision is implementation-specific. This decision should be carefully evaluated by your entire internal implementation team.
- If you wish to continue to use Organization visibility, you must make changes to the business component, view, and picklist properties in order to use Organization visibility.
 - Business Components with Catalog Visibility
 - Applet picklists with Auto Query Mode or New Query set to `None`.
 - Views with Catalog Visibility

After the upgrade, the following business components, views, and picklists—all of which use catalog visibility as the default configuration in Release 7.x—need to be carefully evaluated if they are used in your implementation.

Business Components

The following business components use Catalog as the default popup visibility type:

- Competitor
- Component Product
- Internal Product
- My Competitor
- My Internal Product
- Recommended Product
- Related Issue

- Sales Tool
- Solution

To change the visibility properties

- 1** Log into Siebel Tools as a valid user.
- 2** Choose the Types tab.
- 3** Choose the Business Component object in Object Explorer.
- 4** Scroll across the list applet to find the property Popup Visibility Type and change this value from Catalog to Organization or to another valid property.

Picklists That Start in Query Mode

The following picklists, by default, start with Auto Query Mode set to *New Query* or *None*:

- Asset Mgmt - Internal Product Pick Applet
- CPG Internal Product Pick Applet
- Catalog Admin Category Assoc Applet—Competitor
- Catalog Admin Category Assoc Applet—Literature
- Catalog Admin Category Assoc Applet—Product
- Catalog Admin Category Assoc Applet—Resolution Item
- Catalog Admin Category Assoc Applet—Solution
- Internal Product Pick Applet (eSales)
- Issue Assoc Applet
- Opportunity Management—Product Number Pick Applet
- Opportunity Management—Product Pick Applet
- Order Entry—Line Item Product Pick Applet
- Product Number Pick Applet

- Product Pick Applet
- Product Pick Applet—No Clear
- Product Pick Applet—No Insert
- SR Internal Product Pick Applet
- Sales Tool Pick Applet
- Solution Create List Applet

To change the visibility properties

- 1** Log into Siebel Tools as a valid user.
- 2** Choose the Types tab.
- 3** Choose the Applet object in Object Explorer.
- 4** Scroll across the list applet to find the property Auto Query Mode and change this value from `New Query` or `None` to no value.

To change the Auto Query Mode property to have no value, remove any search specification that would otherwise be inherited from the Business Component.

Views

The following views use Catalog as the default visibility applet type:

- Competitive Company Across Catalogs
- Products across Catalogs
- Sales Tools Across Catalogs
- Service Solution List View (SCW)
- Solutions Across Catalogs

To change the visibility properties

- 1 Log into Siebel Tools as a valid user.
- 2 Choose the “Types” tab.
- 3 Choose the View object in Object Explorer.
- 4 Scroll across the list applet to find the property “Visibility Applet Type” and change this value from `Catalog` to `Organization` or to another valid property.

For analysis and recommendations for choosing and implementing access control strategies from an upgrade perspective, please refer to Siebel SupportWeb for the *Access Control Upgrade and Migration Guide* technical note.

Resolving Duplicate EIM Mappings

Any custom Enterprise Integration Manager (EIM) mappings that were added to previous versions of Siebel eBusiness Applications will be automatically migrated to the new custom repository. If custom mappings are duplicated by new mappings added as part of Siebel version 7.x, errors will result when you use EIM.

NOTE: Custom EIM mappings are only added to old `_IF` tables, not to new EIM tables. If you need to use the new EIM tables, you must manually add the customization to them.

The Siebel Software Configuration Utility that you ran in [“Installing and Configuring the Siebel Database Server Software” on page 109](#) generated a file called `mapclash.txt`. This file lists any custom EIM mappings that conflict with Siebel Systems-provided mappings. The file contains the columns illustrated and defined in [Table 22](#).

Table 22. Mapclash.txt File Column Names and Definitions

Column Name	Definition
IT_Name	Interface table name (example: <code>S_ACCOUNT_IF</code>)
DT_Name	Destination table (base table) name
DC_Name	Destination column (base table column) name

Before proceeding with the upgrade, use Siebel Tools as described below to delete any custom mappings listed in this file.

To delete duplicate custom EIM mappings

- 1** Launch Siebel Tools and select New Customer Repository.
- 2** From the Object Explorer, choose EIM Interface Table > EIM Table Mapping.
- 3** From the EIM Table applet, select the interface table name (*IT_Name*), for example, *S_ACCOUNT_IF*, and query for the interface table.
- 4** From the EIM Table Mappings applet, choose the name assigned for the destination table (*DT_Name*); for example, *S_ORG_EXT_X*.
- 5** From the Object Explorer, choose EIM Table Mapping > Attribute Mapping.
The fourth column in this applet is the *DC_Name*.
- 6** Delete the duplicate mappings.

Safeguarding the New Custom Repository Export File

The Siebel Software Configuration Utility exports the new customized repository to a file called *custrep.dat*, located in the *DBSRVR_PLTFRM_ROOT* directory on the Siebel Server from which you ran the upgrade. Locate this file and make a backup copy of it. You will need it for the upgrade of the production database.

If you make *any* changes to your custom repository after running the development upgrade Siebel Software Configuration Utility, including reapplying custom extensions or modifying EIM mappings in the two previous steps, you must export a new copy of the repository before performing the production upgrade. Use the script *DBSRVR_PLTFRM_ROOT/master_exprep.ucf* to export your current repository to the file that you designated as the output file in the Siebel software configuration utility.

The *master_exprep.ucf* calls the *repimexp* utility, which connects to two separate repositories—the repository whose structure is to be extracted, and the repository whose content is to be extracted. Apart from unusual circumstances, you will always extract *both* the structure and content of the same repository.

For information on exporting repositories, see the “Repository Management Tools” chapter of *Siebel Tools Reference, MidMarket Edition*.

NOTE: Before you export the repository, you should make sure that all Siebel Tools projects are unlocked to promote a stable environment.

Producing a New Custom Configuration File

The repository merge process merged your configuration changes into a new custom configuration repository. You must now compile a new Siebel repository file (.srf) from this repository.

NOTE: If you intend to deploy your Siebel repository file (SRF) in more than one language, set the Tools Language setting and compile an SRF for each language. For information about how to compile an SRF for a specific language, see *Global Deployment Guide, MidMarket Edition*.

To compile a new .srf file

- 1 Using a new Siebel Tools development workstation, connect to the database against which you ran the merge, and then open your new customized repository by choosing File > Open Repository.
- 2 Select Siebel Repository.
- 3 Choose Tools > Compile Project.
- 4 Verify or add the following information in the Object Compiler dialog box:
 - Select All Projects.
 - The name of the output Siebel Repository (.srf) file (the name for your custom configuration .srf file).
- 5 Click Compile.

After the compilation process has finished, distribute the new .srf file to each of your development Siebel Client workstations. For information about how to distribute a new .srf file, see *Siebel Anywhere Administration Guide, MidMarket Edition*.

Regenerating the Database Template File

Following the upgrade, you must regenerate the SQL Anywhere template database file used by Siebel Remote. This process updates its schema to the same version as the database server. You will use the Generate New Database component from a new Siebel Server to do this.

For instructions on regenerating the SQL Anywhere Template file, refer to *Siebel Remote Administration Guide, MidMarket Edition*.

Extracting Developers or Siebel Tools Clients

Extract all Siebel Tools developers and clients using the Database Extract component from a new Siebel Server.

After you have extracted and initialized the mobile databases of all Siebel Tools developers, they must check out a read-only copy of all projects into the local database. For further instructions on setting up Siebel Tools developers, see *Siebel Tools Reference, MidMarket Edition*.

Testing the Upgraded Configuration

You must test the upgraded configuration thoroughly to make sure that all functionality works as expected. You need to execute your full development or acceptance test plan, including at least one synchronization session from a mobile client, before proceeding to upgrade your production environment.

If you make further changes to your customized configuration, remember to re-export the repository to the `custrep.dat` file before running the production environment upgrade.

NOTE: Save the log files generated during the repository upgrade and schema upgrade. Locate the `custrep.dat` file in the `DBSRVR_ROOT\COMMON` subdirectory. Save the `custrep.dat` file for your production upgrade.

Validating the Upgrade

Run the following utilities to verify that your upgrade was successful.

- **DBCHCK.** This utility verifies that the physical schema is in sync with the logical schema (dictionary). If tables are missing or mismatched in either the logical (dictionary) or physical schema, they are reported in the dbchk.log file generated by DBCHCK.
- **DICTUTL.** This utility verifies that all dock object and rule definitions are correct.

To verify that the physical schema is in sync with the logical schema

- 1 Navigate to the Windows DOS command window.
- 2 Type the following command from *SIEBEL_ROOT/bin*:

```
dbchck /S <ODBC_DATASOURCE> /U <USERNAME> /P <PASSWORD> /T  
<TABLEOWNER> /R <REPOSITORY> /L <LOGFILE> /D /A
```

- 3 Review the dbchk.log file for unacceptable errors.
 - Unacceptable errors may occur if data types are mismatched.
 - Acceptable errors may occur if a schema object (tables, indexes) is intentionally external to the repository.
 - Pay attention to schema object (tables, indexes) that are reported in the repository and are not in the physical schema. Some objects may need to be reapplied to the physical schema if they are not intentionally external to it.

To verify that all dock object and rule definitions are correct

- 1 Navigate to the Windows DOS command window.
- 2 Type the following command from *SIEBEL_ROOT/bin*:

```
dictutl /C <ODBC_DATASOURCE> /U <USERNAME> /P <PASSWORD> /D  
<TABLEOWNER> /N <REPOSITORY_NAME> /A y 2> LOGFILE.log
```

Where *LOGFILE* is the log file that you designate for dictutl.

- 3 Review the *LOGFILE.log* file.

Run Statistics

If your RDBMS is DB2 UDB or Microsoft SQL, you should run full statistics on all tables used in your environment.

- If your RDBMS is DB2 UDB, run `updatestats.sql`, found under `DBSERVER_ROOT/DB2UDB`.

Reorganize the P1 Index for DB2 UDB

After your upgrade to Release 7.x, you need to verify that the table `S_EVT_ACT` is reorganized on the P1 index. This happens because, in the repository, the append mode flag is set. If `S_EVT_ACT` is not reorganized by the P1 index, then Siebel Remote performance will be negatively affected.

Reset Database Server Configuration Parameters

After you complete your upgrade, you need to reset your database server configuration to installation settings. See *Siebel Server Installation Guide* for the operating system you are using.

Clean Up Your Repository Files

After you resolve all merge conflicts and errors, export your Prior Customer Repository as a backup for your production environment upgrade. You can use Siebel Tools to safely delete the following repositories:

- Prior standard repository (version of 5.x, 6.x., or 7.0.x)
- New standard repository (version 7.5)

Additional Postupgrade Tasks for Specific Applications

This section describes additional postupgrade tasks and considerations for the following applications:

- Call Center
- eBusiness Application Integration (EAI)
- Quotes
- Workflow

Call Center

NOTE: If you are upgrading from Release 7.0.x to Release 7.5, you do not need to perform this procedure.

During the upgrade to Release 7.x, employee and contact data were moved to the same tables. As a result of this change, your database will contain duplicate logins for Contact user login names.

In order to prevent the creation of duplicate logins which may occur due to this change, the Release 7.x upgrade will append the `ROW_ID` to duplicate logins. You need to resolve your user logins after the upgrade, or users may be unable to log in. For example, contact user login names will appear concatenated with their row ID.

To locate user logins that require resolution of duplicates

- 1** Open your Call Center application and navigate to Site Map > User Administration.
- 2** For each User Administration view (Employees, Persons, and Users) query the User ID field for login name = * + *. This query will bring up all names that are appended with “ + <ROW_ID> ”.

Repeat this procedure for each User Administration view (Employees, Persons, and Users).

Enterprise Application Integration (EAI)

If you use Enterprise Application Integration (EAI), perform the following procedure to update the definitions of the Business Objects to account for changes in data type, length, edit format, or other properties during upgrading to a new version of Siebel Applications.

To upgrade integration objects

- 1** Synchronize the integration objects.
- 2** Validate the integration objects.
- 3** If you receive validation errors, inactive the user keys or fields that cause the error.
- 4** If you receive the error “List Of” in the XML Parent Element, manually remove the value “List Of” from the XML Parent Element.

Quotes

After upgrade, you need to run the following statement to fix an issue where quote items had trailing spaces added accidentally. This script is mandatory.

```
update S_QUOTE_ITEM
setROW_ID = rtrim(ROW_ID)
,   ROOT_QUOTE_ITEM_ID = rtrim(ROOT_QUOTE_ITEM_ID)
,   PAR_SQ_ITEM_ID = rtrim(PAR_SQ_ITEM_ID)
,   PORT_VALID_PROD_ID = rtrim(PORT_VALID_PROD_ID)
,   PROD_PORT_ID = rtrim(PROD_PORT_ID)
,   INTEGRATION_ID = rtrim(INTEGRATION_ID)
;
commit
;
```

Siebel Workflow

Perform the following postupgrade steps after a successful upgrade of Siebel Workflow to Release 7.5.

- Change inbound workflows that contain a “String” type process property to pass the value into type `Binary`; otherwise, the workflow will give the following error message:

```
Output argument '<Value>' in step 'Read from File' contains data
that cannot be passed to string type property 'InputXML'. Data
type: 'MEMBLOCK'; String representation of data body: '<?xml
version="1.0" encoding="UTF-8"?><?'
```

- After an upgrade from Release 6.x to 7.5, manually change the name of the “EAI MQSeries Transport” business service to the name “EAI MQ Series Server Transport”; otherwise, the workflow will give the following error message:

```
[1] Unable to create the Business Service 'EAI MQSeries Transport'
[2] Could not find 'Class' named 'EAI MQSeries Transport'. This
object is inactive or nonexistent.
```

Upgrading to RC2 Encryption

CAUTION: If you have data that was encrypted using the Release 6.x or 7.0.x standard encryptor, and you need to be able to read this encrypted data in Release 7.5, you must upgrade your encryption method after successfully upgrading but before deploying the application.

Release 7.5 delivers a new default encryption method that is based on the RC2 standard. The previous default encryption method (called the standard encryptor) is no longer supported, and data that used the standard encryptor will not be read properly by 7.5 applications unless you upgrade your encryption method to RC2. Use the Encryption Upgrade Utility to convert unencrypted data and data that was encrypted using the standard encryptor to the RC2 encryption method.

CAUTION: Using a non-RC2 encryption method in a Unicode environment will result in irrecoverable data loss.

Perform the following procedures to upgrade your encryption method:

- Verify that all prerequisites are met. See [“Prerequisites” on page 196](#).
- Make sure that the input file includes every column that you want to upgrade. See [“Modifying the Input File” on page 196](#).
- If you customized business component fields to use the old encryption method, verify that you have the correct user property definitions. See [“Changing User Properties” on page 197](#).
- Run `keydbmgr.exe` to change the password or add a new key to the database. See [“Changing the Password or Adding a New Key to a Database” on page 198](#).
- Determine which encryption you are going to use: 128-bit or 56-bit encryption.
 - If you are upgrading to 128-bit encryption, follow the instructions for upgrading to 128-bit encryption, [“To upgrade to 128-bit encryption” on page 199](#) then continue with the upgrade to 56-bit encryption.
 - If you are upgrading to 56-bit encryption, see [“To upgrade to 56-bit encryption” on page 200](#).

Prerequisites

In order to upgrade to the RC2 encryption method, the following prerequisites must be fulfilled:

- The Gateway Server and Siebel Server are installed.
- The repository has been upgraded to the Release 7.5 schema, so that a new column has been created to store the key index for the encrypted column.
- If you created or customized columns to use the old encryption method (the standard encryptor), for each encrypted column that you want to upgrade, you need to create a new column to store the key index.
- The key database (`keyfile.bin`) must already exist. (A default keyfile was created in the `SIEBSRV_ROOT/siebsrvr/admin` directory when you installed the Siebel Server.)
- The password must be stored in the database.

Modifying the Input File

The input file `encrypt_columns.inp` indicates the table and column that store the encrypted data, and the table and column that store the key index. The input file is located in `DBSRV_ROOT/DBSRV/bin` directory. If you wish to execute the utility from the command line, place this file in the `SIEBSRV_ROOT/bin` directory.

The input file must include every column that you want to upgrade. The first line of the input file indicates a table name with brackets around it. The table name should be followed on subsequent lines by all the columns that will be upgraded for that table. Each column requires a table column to store the key index, so this is specified after the column name; for example:

```
[TABLE_NAME]
COLUMN_NAME TABLE_NAME_FOR_KEY COLUMN_NAME_FOR_KEY
```

After each table, skip a line, and continue with subsequent tables. Here is a sample input file:

```
[S_ORDER]
CC_NUMBER S_ORDER CCNUM_ENCRPKEY_REF
```

```
[S_DOC_ORDER]
CC_NUMBER S_DOC_ORDER CCNUM_ENCRPKEY_REF

[S_PER_PAY_PRFL]
PAY_ACCNT_NUM S_PER_PAY_PRFL CCNUM_ENCRPKEY_REF
```

To support upgrade of non-encrypted field to RC2 encryption, add the letter **N** to the end of the column; for example:

```
[S_NEW_TABLE]
NAME S_NEW_TABLE NAME_KEY_INDEX N
```

Changing User Properties

If you customized business component fields to use the old encryption method (the standard encryptor), make sure that your custom buscomp field user properties are defined with the values provided in the table below. (An example is provided for the Quote business component.)

NOTE: By default, data encrypted using the old encryption method will use `ROW_ID` as the Encrypt Key Field. You may need to create a calculated field on each business component for the Encrypt Read Only Field.

Field User Property	Value	Example for the Quote BusComp
Encrypted	Y	Y
Encrypt Service Name	RC2 Encryptor	RC2 Encryptor
Encrypt Key Field	<Key Index Field>	Credit Card Number Key Index
Encrypt Read Only Field	<Read Only Field>	Credit Card Number Read Only

Changing the Password or Adding a New Key to a Database

If you need to change the password or add a new key to your database, perform the following steps:

To change the password or add a new key

- Run `keydbmgr.exe` to change the password or add a new key to your database. From `SIEBSVR_ROOT/bin`, enter the following command:

```
keydbmgr.exe /U USERNAME /P PASSWORD /C  
CONFIGURATION_FILE_NAME /L LANGUAGE
```

where:

- `USERNAME` = user name
- `PASSWORD` = password
- `CONFIGURATION_FILE_NAME` = name of the configuration file (the default is `siebel.cfg`). Make sure that the configuration file is pointing to the correct database.
- `LANGUAGE` = The base language for your installation (the default is `enu`)

for example,

```
keydbmgr.exe /u sadmin /p PASSWORD /c siebel.cfg /l enu
```

NOTE: Use the default password `kdbpass` to login and change the password.

Upgrading Your Encryption Method

Follow the instructions below for the encryption you are going to use: 56-bit encryption or 128-bit encryption.

- If you purchased the Strong Encryption Pack for 128-bit encryption, follow the instructions, [“To upgrade to 128-bit encryption” on page 199](#), and then continue with the upgrade to 56-bit encryption.

NOTE: Verify encryption requirements and constraints for your deployment before you upgrade your encryption to 128-bit.

- If you will perform the standard encryption upgrade to 56-bit encryption, skip the upgrade to 128-bit encryption, and go directly to the instructions [“To upgrade to 56-bit encryption” on page 200](#).

To upgrade to 128-bit encryption

- 1 Install the Strong Encryption Pack that you purchased separately.
- 2 Run `keydbupgrade.exe` to upgrade to 128-bit encryption.
From `SIEBSRV_ROOT/bin`, enter the following command:

```
keydbupgrade.exe /U USERNAME /P PASSWORD /C  
CONFIGURATION_FILE_NAME /L LANGUAGE
```

where:

- `USERNAME` = user name
- `PASSWORD` = password
- `CONFIGURATION_FILE_NAME` = name of the configuration file (the default is `siebel.cfg`)
- `LANGUAGE` = The base language for your installation (the default is `enu`)

for example,

```
keydbupgrade.exe /u sadmin /p PASSWORD /c siebel.cfg /l enu
```

- 3 Continue by upgrading to 56-bit encryption.

To upgrade to 56-bit encryption

- 1 Verify that the input file `encrypt_columns.inp` includes all the columns that you want to upgrade. If necessary, review [“Modifying the Input File” on page 196](#).
- 2 Run `encryptupg.exe` to upgrade to 56-bit encryption.
From `SIEBSRV_ROOT/bin`, enter the following command:

```
encryptupg.exe /U USERNAME /P PASSWORD /C ODBC_CONNECT_STRING  
/D TABLEOWNER /J INPUT_FILE /N REPOSITORY_NAME /K KEY_FILE_NAME  
/L LOG_FILE
```

where:

- `SIEBSRV_ROOT` = root directory of your Siebel Server installation
- `USERNAME` = user name for the database
- `PASSWORD` = password for the database
- `ODBC_CONNECT_STRING` = ODBC connect string for the database
- `TABLEOWNER` = table owner for the database
- `INPUT_FILE` = the name of your input file (the default is `encrypt_columns.inp`)
- `REPOSITORY_NAME` = repository name (the default is Siebel Repository)
- `KEY_FILE_NAME` = the absolute path to the key file (the default location is `SIEBSRV\admin\keyfile.bin`)
- `LOG_FILE` = the default log file is `encryptupg.log`

NOTE: If you have custom encrypted fields, you can validate that all business component fields are pointing to the same column by appending the command with `/v y`. For example,

```
encryptupg.exe /U USERNAME /P PASSWORD /C ODBC_CONNECT_STRING /D  
TABLEOWNER /J INPUT_FILE /N REPOSITORY_NAME /K KEY_FILE_NAME /L  
LOG_FILE /v y
```

But, some business components may not use the same column intentionally.

- 3 Repeat [Step 2 on page 200](#), `encryptupg.exe`, for each additional database. For each database, use the appropriate user name, password, ODBC connect string, and table owner.
- 4 After the upgrade is complete, compile a new Siebel repository file. See [“Producing a New Custom Configuration File” on page 188](#).

Troubleshooting

If you fail to change user properties, you may get one or more of the following repository validation error messages. Examples of possible errors and sample remedies are provided below.

Possible Error: Validation failed for Personal Payment Profile buscomp, Account Number field does not have field user property Encrypt Key Field defined or activated.

Sample Fix: Go to the Personal Payment Profile business component, Account Number field, and verify that the following field user properties exist:

Encrypted = y

Encrypt Key Field = `<KEY_INDEX_FIELD>`

Encrypt Service Name = RC2 Encryptor

Postupgrade Tasks for Global Deployments

The following procedures are for global deployments, deployments that intend to become global in the future, and deployments that wish to take advantage of global deployment features.

- [Setting Up Your Environment to Support Global Time Zone](#)
- [Migrating to Unicode](#)

Thoroughly read *Global Deployment Guide, MidMarket Edition* before you internationalize any environment.

Setting Up Your Environment to Support Global Time Zone

Global deployments typically span multiple geographies and have users working in several different time zones. The global time zone feature converts and stores date and time data using the Universal Time Coordinated (UTC) standard, which is equivalent to Greenwich Mean Time, but without daylight savings time. This feature allows you to track dates and times in a common format across time zones.

NOTE: Although enabling your environment for global time zone is optional in Release 7.x, it is strongly recommended that you operate your production environment with global time zone enabled.

CAUTION: If you previously upgraded to Release 7.0.x and you enabled your environment for global time zone, you do not need to repeat the same procedure. Instead, you need to upgrade data from columns that were *not* UTC-enabled in Release 7.0.x that are UTC-enabled in Release 7.5.

Perform one of the following procedures, as appropriate to your upgrade path:

- **For upgrades from Release 6.x to 7.x.** See [“Enabling Global Time Zone” on page 203](#) for information about enabling global time zone support after an upgrade.
- **For upgrades from Release 7.0.x to 7.5.** See [“Upgrading UTC Delta Columns” on page 203](#) for information about upgrading 7.0.x data for columns that were *not* UTC-enabled in Release 7.0.x that changed to UTC-enabled in Release 7.5.

Enabling Global Time Zone

To enable global time zone support, after your upgrade is complete, you need to set the global time zone parameter (Universal Time Coordinated system preference) to `TRUE` through Siebel Tools. The UTC system preference is not enabled after any upgrade to Release 7.x. To enable the global time zone feature, you will need to run the UTC conversion utility. The high-level steps you need to perform are outlined below (see *Global Deployment Guide, MidMarket Edition* for complete instructions).

To enable global time zone support after an upgrade

- 1** Stop Application Servers.
- 2** Set UTC system preference to `FALSE` before you convert your historical data.
- 3** Prepare your data for conversion to global time zone, as instructed in *Global Deployment Guide, MidMarket Edition*.
- 4** Convert your historical data, using the UTC conversion utility, to make all existing date/time values consistent with global time zone logic. Perform this step as instructed in *Global Deployment Guide, MidMarket Edition*.
- 5** After you have confirmed that your UTC conversion was successful, turn on global time zone by setting the UTC system preference to `TRUE`. To reset this parameter through Server Manager, navigate to Application Administration > System preferences.
- 6** Bring up Application Servers and Web Server.

Upgrading UTC Delta Columns

If, before upgrading to Release 7.5, you previously upgraded to Release 7.0.x and you already ran the UTC conversion utility, perform the following steps to upgrade the delta columns (those that were not UTC-enabled in 7.0.x but are UTC-enabled in 7.5).

CAUTION: Back up your database before you run the UTC delta conversion procedure.

To upgrade 7.0.x data for columns that have changed to UTC-enabled in 7.5

- 1 Open the `master_utc.ucf` file from `DBSRVR_ROOT\DATABASE_PLATFORM\master_utc.ucf`, and edit the File Name parameter by replacing `driver_utc.ucf` with `driver_utc_delta.ucf`.
- 2 Launch the Siebel Software Configuration Utility by selecting Start > Programs > Siebel Enterprise Servers 7.0 > Configure DB Server.

You can also launch the utility from the command prompt by typing:

```
ssincfgw -l language_code -v Y
```

where `language_code` is the three-letter code (all capitals) for the language in which you want to display the GUI.

For example, to run the UTC Upgrade Utility in English, you would type:

```
ssincfgw -l ENU -v Y
```

NOTE: If you are prompted to select an `.scm` file, select `dbsrvr.scm`.

- 3 When you reach the Database Server Options screen, select Run Database Utilities.
- 4 At the Database Utility Selection screen, select UTC Upgrader.
- 5 Continue until you are prompted to run the Siebel Upgrade Wizard (`siebug`), then click cancel.

For more information about UTC, see *Global Deployment Guide, MidMarket Edition*.

Migrating to Unicode

CAUTION: Verify that your encryption method has been upgraded before you proceed with the Unicode migration. Use of a non-RC2 encryption method in a Unicode environment might result in irrecoverable data loss. See [“Upgrading Your Encryption Method” on page 199](#).

After you have upgraded your Siebel eBusiness Application to Release 7.5, you can migrate your upgraded database from a non-Unicode code page (or character set) to Unicode.

NOTE: Migration to Unicode is optional and requires the assistance of Siebel Expert Services or Siebel Professional Services.

For information about Unicode and global deployment for Siebel eBusiness Applications, see *Global Deployment Guide*.

To migrate to Unicode, you perform a two-step process

- 1** Upgrade your non-Unicode code page database to Release 7.5.
- 2** Migrate your upgraded Release 7.5 database from a non-Unicode code page to Unicode. To perform the migration, you can use either database vendor native utilities or Siebel utilities. For more information, contact Siebel Expert Services or Siebel Professional Services.

CAUTION: You cannot use a non-Unicode code page for your development environment and then later migrate to Unicode for your production environment.

Planning Considerations for the Unicode Migration

If you are planning to migrate your upgraded application to Unicode, you need to consider the following points:

Migration to Unicode will increase the size of your database. For this reason, you need to allocate additional space for your database before migrating to Unicode. For more information, contact Siebel Expert Services or Siebel Professional Services.

Migration to Unicode may cause truncation of certain data in DB2 UDB databases. In the past, long columns with a type of varchar could have a maximum length of 16,383 characters. However, in Unicode, the maximum length of long columns with a type of varchar is 16,350. During the migration to Unicode, long columns of type varchar that exceed 16,350 will be truncated. To prevent this, you can perform tasks to identify which data may be truncated and take appropriate measures before migration. For more information, contact Siebel Expert Services or Siebel Professional Services.

DDLICT does not preserve custom table space information for DB2 UDB databases. This will present a problem during your migration to a Unicode code page, because you need to know which tables need to be recreated. If your RDBMS is DB2 UDB and you use custom table spaces, you need to modify the upgrade scripts to handle custom table spaces.

NOTE: Contact Siebel Expert Services or Siebel Professional Services for instructions about how to modify upgrade scripts to handle custom table spaces.

Migration to Unicode may affect integration with third party systems. For more information, contact Siebel Technical Support or Siebel Expert Services.

Release 7.5 supports two types of Unicode:

- **UTF-8.** UTF-8 is supported for Oracle databases for Release 7.5. UTF-8 uses the same encoding for Western European languages.
 - Occupies one byte for Western European languages.
 - Occupies up to three bytes for some Asian languages, such as Japanese.
- **UCS-2.** UCS-2 is supported for DB2 UDB and Microsoft SQL Server databases for Release 7.5. UCS-2 does not map one-to-one with Western European languages, but it occupies two bytes for all languages.
 - Occupies two bytes for Western European languages.
 - Occupies two bytes for Asian languages.

Migrating the Client Configuration to the Web Client

6

This chapter describes procedures you need to perform to migrate to the new Release 7.x Siebel Web Client.

NOTE: If you are upgrading from Release 7.0.x to Release 7.5 (that is, if you previously upgraded to Release 7.0.3 or 7.0.4 and you already performed this migration), you do not need to repeat the procedures described in this chapter.

About Migrating to the Siebel Web Client

In Release 7.x, the Siebel Web Client is used as the deployment framework for all Siebel eBusiness Applications. The Siebel Web Client is a zero-footprint client that provides high levels of functionality and interactivity without requiring persistent client-side software installation. Three areas need additional attention as part of the migration to the Release 7.x Web Client. These are:

- Customized applets and views
- Scripts on UI Objects and with references to UI constructs
- Client-side interfaces to external desktop applications

The Release 7.x Application Upgrader includes a new Web Client Migration Wizard that helps to upgrade customized view and applet definitions used by the Siebel Dedicated Client and Windows Thin Client in previous Siebel releases to the new Web Client. This wizard is run after the repository has been upgraded.

The Web Client Migration Wizard supports a batch mode that converts multiple view or applet definitions at a time, using customizable models to automatically bind the user interface controls to Web Client templates. While this process is automated, you may want to make further manual adjustments to the templates produced to make sure that your customized interface takes full advantage of the interactivity and functionality of the Release 7.x Web Client.

NOTE: This process is only required for views or applets customized in previous Siebel releases, because the Release 7.x Web Client includes standard versions of all applets and views from previous Dedicated Client releases.

The Web Client Migration Wizard will not migrate Siebel VB or Siebel eScript scripts attached to applets. These scripts will either need to be moved to the Server or to be rewritten in JavaScript, the Web Client-supported language for user interface scripting.

The Release 7.x Web Client supports most of the events and methods on user interface objects supported by previous Siebel releases. In addition, the Web Client provides comparable inbound and outbound client-side integration capabilities to what was available in the Business Object Interfaces in previous releases of the Siebel Dedicated Client.

The remainder of this chapter outlines the migration process and provides detailed instructions for performing migration tasks.

Migration Approach

This section provides an overview of migration tasks. Detailed migration tasks are described later in the chapter.

NOTE: If you are upgrading from Release 7.0.x to Release 7.5, (for example, from Release 7.0.3 or 7.0.4 to Release 7.5), you do not need to perform the migration described in this chapter.

Applets and Views

In Release 7.x., applets and views in the user interface are rendered using combinations of object definitions stored in the Siebel repository and layout and formatting information contained in applet Web templates. Objects such as controls and list columns are mapped to placeholders in Web templates. At run-time, the Siebel Web Engine creates these objects, places them in the appropriate spot in the Web template, retrieves relevant data from the database, and uses HTML contained in the Web template to display the applet in the user interface.

All the standard applets and views that are shipped as part of the Release 7.x eBusiness Applications have Web layouts. However, if you have made customizations to applets and views, you need to migrate these customizations to the Web client. Siebel provides the Web Client Migration Wizard and the Web Layout Wizard to help you perform these migrations by converting UI elements to the Release 7.x Web Client.

NOTE: Siebel Systems recommends that you use the Web Client Migration Wizard or the Web Layout Wizard to migrate your new and customized applets and views to the Web client.

The following types of applet and view customizations need to be migrated:

- New applets
- New views

- Customized applets (applets to which you have added or deleted controls or list columns)
- Customized views (views to which you have added or deleted applets)

NOTE: For detailed migration steps, see [“Migrating Applets and Views” on page 216](#).

Migration Wizard Benefits

Using the migration wizards can benefit you and your organization in many ways. These benefits include:

- Standardization of display for applets and views
- Minimization of the number of templates required and the cost of creation and maintenance
- Simplification of application-wide modification and upgrade

Scripts

For upgrades to Release 7.x, some scripts on UI objects need to be migrated to Browser script or Server script.

Scripting in the Web Client

In Release 7.x, there are two basic types of scripts you can write to add procedural logic to an application configuration – Server scripts and Browser scripts.

Server Scripts

Server scripts execute in the Object Manager. They existed in prior releases and continue to be supported in the Release 7.x. These scripts are written in Siebel VB and Siebel eScript .

The following are the various types of server scripts:

- Business Component Server Script
- Business Service Server Script

- Application Server Script
- Applet Server Script

These scripts allow developers to script the behavior of Business Components, Business Services, Applications, and Applets, by writing event handlers for the various events exposed by the scripting model.

Browser Scripts

In Release 7.x, there is a new type of script called Browser script that executes in and is interpreted by the Browser. Browser scripts are written in JavaScript and interact with the Document Object Model (DOM) as well as the Siebel Object Model available in the High Interactivity Web Client framework. A developer can script the behavior of Siebel events as well as the browser events that are exposed through the DOM. Siebel Tools allows you to write scripts by selecting the appropriate User Agent. For more information about events that are exposed through the DOM, see *Siebel Tools Online Help, MidMarket Edition*.

NOTE: The DOM for Internet Explorer and Netscape Navigator are different.

Scripts to Be Migrated

All 6.x scripts written on applets will need to be either migrated to Applet Server Scripts or BusComp Server Scripts, or rewritten as Browser scripts. The 6.x scripts on the Application, Business Services, and Business Components will need to be reviewed to identify references to methods, such as UI methods, which cannot be accessed from Server scripts.

NOTE: During an upgrade from Release 7.0.x (7.0.3 or 7.0.4) to Release 7.5, the 7.0.x browser scripts and objects that contain 7.0.x browser scripts will be automatically upgraded to 7.5.

Identifying Scripts to Be Reviewed or Migrated

You can identify the number of 6.x applet scripts to be reviewed or migrated by looking at the Object List Editor in Flat mode in Siebel Tools.

To help locate server side scripts that may need to be modified, in Siebel Tools you can run the Validator on BusComp, Business Service, and Applications for which the “Scripted” flag is set to “True.”

Applet Scripts

Applet script migration falls into the following major areas:

- Certain 6.x applet scripts such as Control_Click event handlers may be moved to Business Components.
- Release 6.x ActiveX Controls that have been scripted will need to be rescripted in Browser script.
- Other 6.x applet scripts need to be rewritten in Browser script.

BusComp and Application Scripts

Release 6.x BusComp and Application scripts that contain references to UI methods need to be modified. For example, calls to MsgBox need to be replaced by calls to the RaiseError/RaiseErrorText method. Other modifications may include moving scripts to Browser script. For more information, see *Siebel Tools Online Help, MidMarket Edition*.

NOTE: For more information about migrating scripts, see [“Migration of Scripts” on page 233](#).

Client-Side Interfaces

In Release 7.x, outbound and inbound COM interfaces are available. However, you may need to perform migration tasks to achieve optimal functionality.

Outbound COM Interfaces

As in previous releases, you can invoke desktop applications, such as Microsoft Excel, from the Siebel client.

In Release 7.x, this functionality is available through Browser scripts on Microsoft Internet Explorer only.

Inbound COM Interfaces

In previous releases, desktop applications communicated with the Siebel Client through an Automation interface in the Windows Client.

In Release 7.x., the High Interactivity Web Client provides similar automation interfaces.

NOTE: For more information, see [“Migration of Client-Side Interfaces” on page 240.](#)

Migration Tasks

This section covers the tasks involved in the three major areas of migration—applet and view migration, script migration, and client-side interface migration.

Exposing Hidden Properties

In Release 7.x, certain properties in Siebel Tools are no longer applicable for the Web Client. By default, these properties are hidden. Before you begin the migration to the Siebel 7.x Web Client, you should expose these properties. Exposing these properties facilitates the migration of your customizations to Release 7.x.

To expose hidden properties

- 1** Using a text editor, open the Tools.cfg file, located in the SiebelToolsDir\bin\<lang> directory.
- 2** Under the [Siebel] section in the Tools.cfg file, locate the parameter titled ClientConfigurationMode.

The default value for ClientConfigurationMode is web, which hides properties that are no longer applicable.
- 3** To expose these attributes for migration purposes, change the value from web to All.
- 4** Save and close the Tools.cfg file.
- 5** Launch Siebel Tools and note the previously hidden properties are once again visible.

NOTE: When you have completed the migration to Release 7.x, it is recommended that you set the ClientConfigurationMode = web to hide properties that are no longer applicable.

Migrating Applets and Views

To transform your customized applets and views to closely corresponding Web client applets and views, you can use either the Web Client Migration Wizard or the Web Layout Wizard. Using the wizards, you select model applets and views to base your customized applets and views on. The model applets and views you select determine the Web layouts, including the layout of controls, of your migrated applets and views.

NOTE: Make sure that the model applets you select contain a sufficient number of Web Template Items to map all items in your customized applets.

The degree of control you want to have over the migration process will determine which wizard you use.

- **Web Client Migration Wizard.** Use the Web Client Migration Wizard if you:
 - Want all your applets of a type to have one Web layout
 - Want all of your views of a type to have one Web layout
 - Have selected one model applet of each type and one model view of each type that meet your needs
 - Want to migrate your applets and views in large batches
- **Web Layout Wizard.** Use the Web Layout Wizard if you:
 - Want to use more than one model applet for each applet type
 - Want to use more than one model view for each view type
 - Want to migrate your applets and views in multiple batches

NOTE: For more information about using the Web Layout Wizard, see *Siebel Tools Reference*.

Model Applets and Views

The Siebel repository contains models for each type of applet and view. These model objects have been designed to support diverse migration needs. For example, the model form applet has the standard Web applet buttons, Web templates, and a large number of controls mapped to Web template placeholders.

Before using model applets and views, you should compare the applets and views you intend to migrate with the structure of the models to ensure that the models suit your needs. If necessary, you can modify model applets and views to meet your requirements. For example, you can modify models to use different Web templates, controls, or control mappings. Alternatively, you can use your own custom Web applets and views with the wizards. If you choose to use your own model applets and views, you should be sure that you fully understand the behavior of the wizards.

CAUTION: If you choose to use your own model applet with the wizards, you must check that the `HTML Sequence` property is defined for all controls in the applet. The `HTML Sequence` property determines the position of a control relative to other controls in an applet, indicating tab sequence. If controls in a model applet do not have `HTML Sequence` numbers defined, the wizards may fail.

You may want to test the wizards against a few applets and views before running the wizards against an entire repository. To do this, see [“Remigrating Applets” on page 229](#) and [“Remigrating Views” on page 231](#).

Out of the box, Release 7.x offers the following model applets and views:

- Model Chart applet
- Model Form applet
- Model List applet
- Model MVG applet
- Model Pick applet
- Model Tree applet
- Model More Applets view

- Model One Applet view
- Model Tree Applet view

The wizards map items from the dedicated client layout to the Web client using the following information from model applets and views:

- **Modes to map Applet Web templates.** Modes determine the kind of actions available in applets to users. For example, some applets are read-only. Other applets allow users to edit data. Typical modes include: Base, Edit, and Edit List. The mode also determines which buttons appear in Web templates. For example, the Edit button appears in applets set to Edit mode, but does not appear in applets set to Base (Read Only) mode. For more information, see *Siebel Tools Reference*.
- **Web templates to use for each mode.** Templates contain placeholders for applets and are used to render Siebel views. The mode selected determines the Web template to be used. For more information about Web templates, see *Siebel Tools Reference*.

NOTE: For Chart and Tree applets, the wizard only uses the Web Template information. For this reason, any applet of the appropriate type may be used as a model applet for Chart and Tree applets.

- **The sequence in which textboxes are mapped.** For form applets, the wizards determine the new relative placement of controls by establishing a placement sequence based on the Top and Left properties of each control and then correlating the placement sequence to Web template item identifiers in the model applet. The process is the same for list applets, except that the wizards determine the placement sequence of the list columns based on their sequence property. For more information, see *Siebel Tools Reference*.
- **The buttons that appear on migrated applets.** Release 7.x Web applets typically contain additional buttons that were not used by their dedicated client predecessors. For example, the Next and Previous buttons used to scroll through record sets in the Web client did not appear in the dedicated client. During migration, the wizards automatically propagate buttons contained in the model applet so that these types of buttons do not require manual work.

NOTE: Siebel Systems recommends that you use the Web Client Migration Wizard to migrate customized and new applets and views.

Applet Migration Examples

The examples provided below illustrate the way in which the wizards use Web Template Items in model applets to create Web layouts that closely correspond to dedicated client applets.

Figure 7 on page 219 shows a custom form applet, the ACME Contact Form Applet, as viewed in the Siebel 2000 Applet Designer. The circled numbers show the placement sequence of each textbox as determined by the Web Client Migration wizard. Since the Top and Left properties of controls were used to determine the placement of controls within a typical Siebel 2000 applet, these properties form the basis of the migration placement sequence. The placement sequence progresses from the upper left corner to the lower right corner by moving from left to right, then top-to-bottom pattern.

Buttons that are on the applet (referenced by the circled numbers in Figure 7) are not migrated the same way as text boxes since the standard Release 7.x Web applet layout typically places these in the applet header. The circumstances in which custom buttons are automatically exposed to the migrated applet are discussed in more detail below.

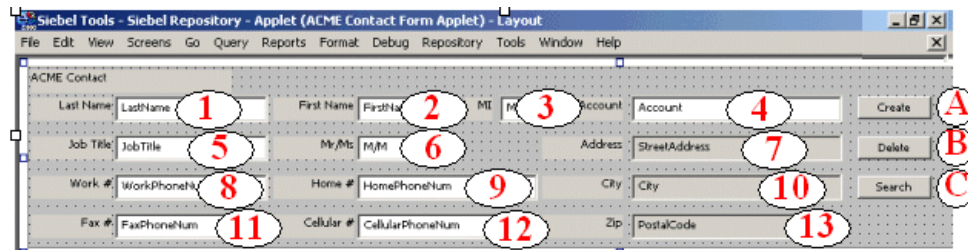


Figure 7. ACME Contact Form Applet in Siebel 2000 Applet Designer

Figure 8 shows a simplified Model Form Applet in the Release 7.x Web Applet Layout Editor.

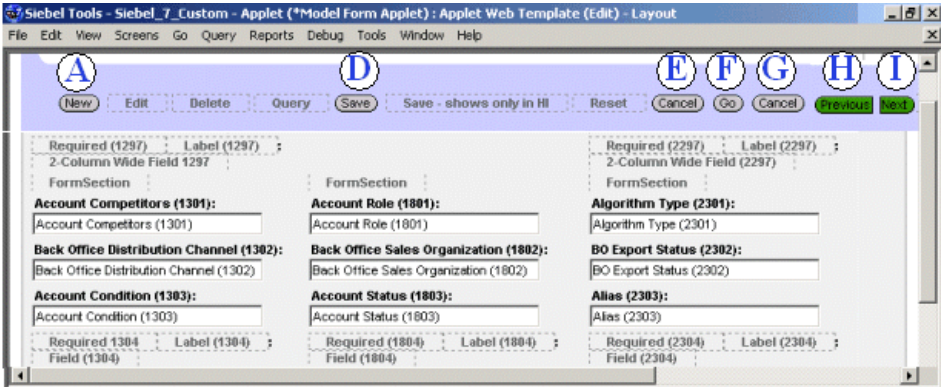


Figure 8. Simplified Model Form Applet

In Figure 8 on page 220, the underlying Web template in the figure has been modified to show the identifiers of the <swe: control> tags used to position controls within the applet. Note that the first text box (Account Competitors) that has been mapped to a <swe: control> tag has an identifier of 1301. The textboxes below it have identifiers of 1302 (Back Office Distribution Channel) and 1303 (Account Condition), respectively. The <swe:control> tag with the identifier 1304 does not have any control mapped to it. The adjacent columns of controls have identifiers that begin with 1801 (Account Role) and 2301 (Algorithm Type). In all, there are nine text boxes that have been mapped in this model form applet with the following identifiers:

Identifier of Applet Web Template Item	Implied Placement Sequence
1301	1
1302	2
1303	3
1801	4

Identifier of Applet Web Template Item	Implied Placement Sequence
1802	5
1803	6
2301	7
2302	8
2303	9

The numeric order of the identifiers forms a logical placement sequence that will be correlated with that of the applet being migrated. The wizards only map controls from the applet being migrated to placeholders that have controls in the model applet.

The model applet also contains a number of buttons referenced in [Figure 8 on page 220](#) by the circled letters D through I. These buttons will be automatically propagated to the migrated applets.

NOTE: In this example, the controls have names that correspond to fields on the Account business component. However, the controls could correspond to fields from any business component and it would not change how the wizards use them. The wizards do not care about names or fields of the controls in the model applet; they only evaluate which <swe:control> tags (placeholders) have controls mapped to them and the numeric value of their identifiers.

Figure 9 shows the results of using the model applet shown in Figure 8 on page 220 to migrate the ACME Contact Form Applet.

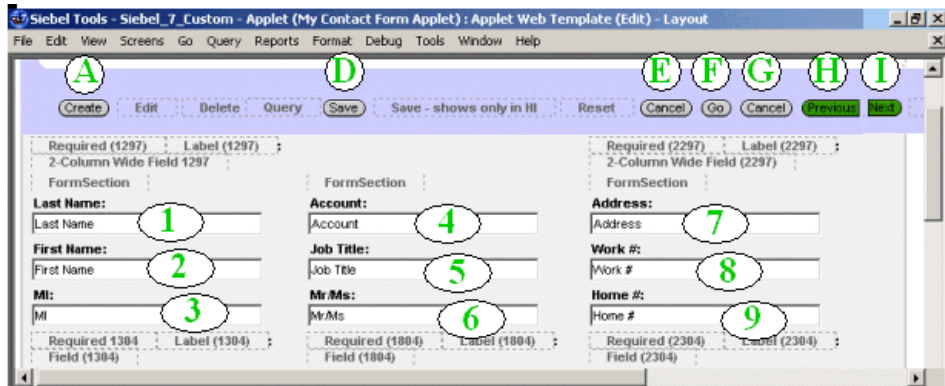


Figure 9. Migrated Acme Contact Form Applet

The circled numbers in Figure 9 correspond to the placement sequence of controls in the pre-migrated applet. The positioning of the textboxes in the migrated applet preserves the general orientation of controls in the original applet where controls with the lowest sequence number appear near the upper left corner and those with the highest appear near the lower right corner.

Figure 9 on page 222 also shows that controls 10 through 13 were not migrated at all; only nine textboxes were migrated because the model applet only had textboxes mapped to nine underlying placeholder tags. To avoid the problems that this can cause, you should be sure to use model applets that have model controls mapped to all the placeholders that you expect your migrated applets to use. This means that model applets tend to have controls mapped to most of the placeholders that are contained by the underlying Web template. However, there are some placeholders, such as those used for 2-Column Wide Fields (for example, placeholders 1297 and 2297 in Figure 8 on page 220 and Figure 9 on page 222) that may be better suited for usage on a case-by-case basis than as a general purpose anchor for controls.

All of the buttons on the migrated ACME Contact form applet, ([Figure 9 on page 222](#)), come from the model applet, ([Figure 8 on page 220](#)). The migrated applet contains buttons D through I, appearing the same as they did in the model form applet ([Figure 8 on page 220](#)); however, the caption property of Button A has changed from New to Create. The original ACME Contact form applet contains a control named NewRecord (button A) as does the model applet. Since it appears in the model applet, the wizard exposes it in the migrated applet. However, its caption property: Create reverts back to the value originally used by the ACME Contact form applet. By contrast, buttons B and C from the original applet are not exposed in the migrated applet because these controls have no corollary in the model.

The transformation between [Figure 7 on page 219](#) and [Figure 9 on page 222](#) also illustrates the results of merging labels and fields, a step that is discussed in [“Preparing the Prior Customer Repository for the Merge” on page 124](#). This step automatically inactivates label controls and writes their caption property to the caption property of the adjacent text box controls. The net effect is that in Release 7.x, a single text box control is responsible for rendering itself as well as the label that appears above it. This simplifies the task of maintaining the controls in an applet and reduces the number of controls required to support the user interface.

The wizards behave similarly when migrating list applets, except that the placement sequence of list columns in the original list applet is determined by the list column sequence property instead of the top and left properties of text box controls.

Figure 10 illustrates the rules for migrating list applets:

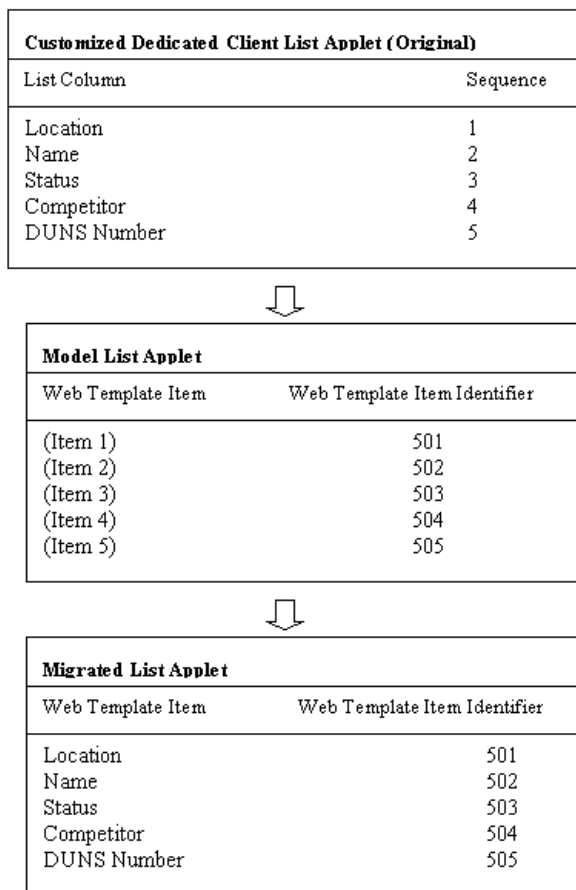


Figure 10. Migration of a Dedicated Client List Applet to a Web Client List Applet

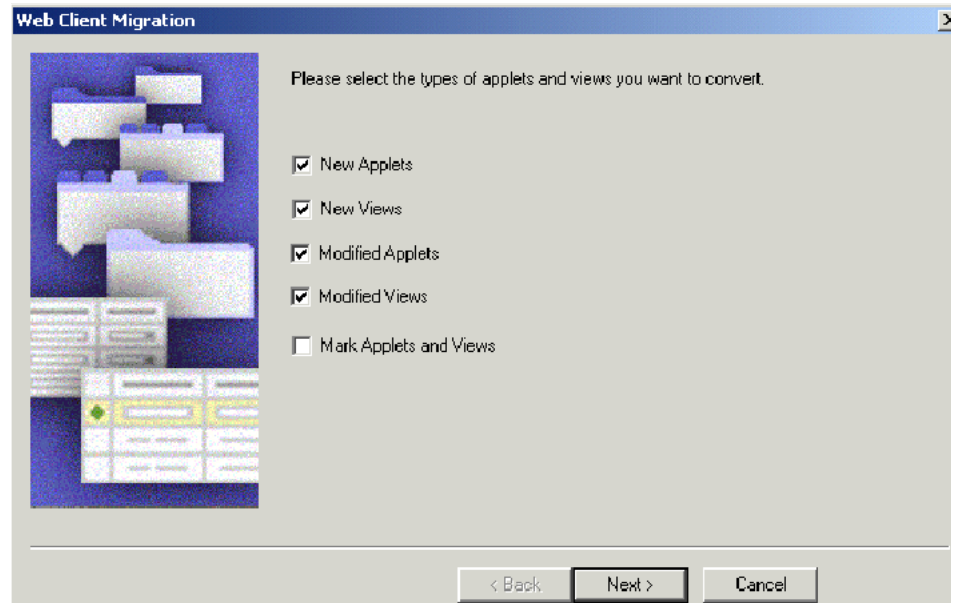
Migrating Customizations Using the Web Client Migration Wizard

The following section describes how to use the Web Client Migration Wizard to migrate your customized and new applets and views.

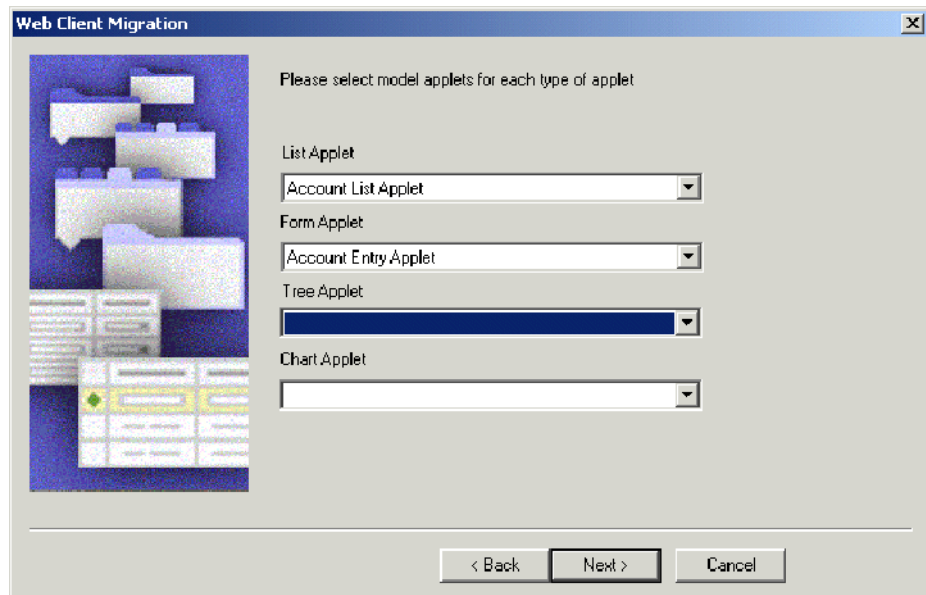
NOTE: For information about using the Web Layout Wizard, see *Siebel Tools Reference*.

To migrate customizations to applets and views

- 1 Once the Application Upgrader has completed, go to the Application Upgrade Object List View in Siebel Tools. In Siebel Tools, select Screens > Application Upgrader > Application Upgrade Object List.
- 2 Invoke the Web Client Migration Wizard. In Siebel Tools select Tools > Upgrade > Web Client Migration.



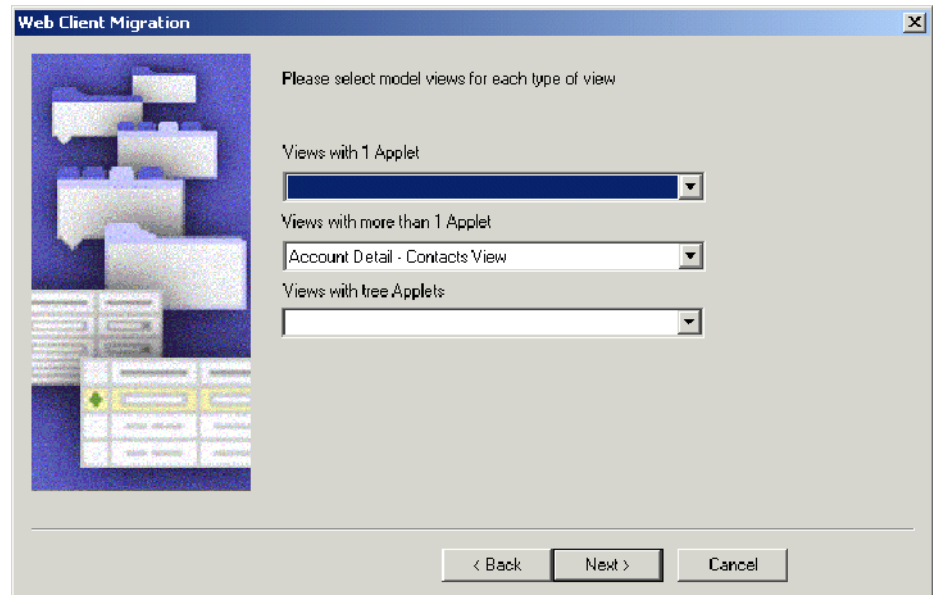
- 3** Select the various types of customizations that you wish to migrate:
 - New applets
 - New views
 - Modified applets
 - Modified views
- 4** At this point, you can choose to either have the wizard perform the migration or just mark the applets and views to be migrated, and then individually migrate the applets and views later.
- 5** On the next screen of the wizard, select model applets for each major applet class—Form, List, Chart, Tree.



- 6 The following screen allows you to choose a model view for each category of view:

- View with one applet
- View with two or more applets
- Views containing tree applets

Any view of the appropriate category may be used as a model view. The information used by the wizard includes the templates to be used and the applet modes to use for the View Web Template Items.



- 7 Run the wizard.

When the wizard has completed, all the applets and views that have been migrated or are to be migrated are indicated. The Comments Field is used for this purpose and the following text is appended to the existing comments:

- Mign. This is appended to the Comments Field of new applets or views that have been migrated

- MigM. This is appended to the Comments Field of modified applets or views that have been migrated
- TBMN. This is appended to the Comments Field of new applets or views to be migrated
- TBMM. This is appended to the Comments Fields of modified applets or views to be migrated

Reviewing Applet Migration

After you have migrated applets, you should review them for accuracy.

To review new applet migration

- 1** In Siebel Tools, go to the Applet List View.
- 2** Search for applets that are marked as “MigN.”
- 3** Sort by Class.
- 4** For each applet in a given class, right-click on the applet. Select Edit Web Layout.
- 5** Review the layout, and make any appropriate changes to the layout using the Applet Web Editor.
- 6** If you notice that the same changes need to be made for a large number of applets, you can run the Web Layout Wizard to remigrate the applets.

Common problems that may require remigration:

- A set of non-Field controls is misplaced or absent in the Web Layout
- The Template needs to be changed
- Field controls are misplaced in the Web layout

NOTE: Changes that do not fall into any of the above categories will need to be implemented manually for each applet.

Remigrating Applets

If a number of applets need to be remigrated, you can run the Web Layout Wizard to remigrate the applets.

To remigrate applets

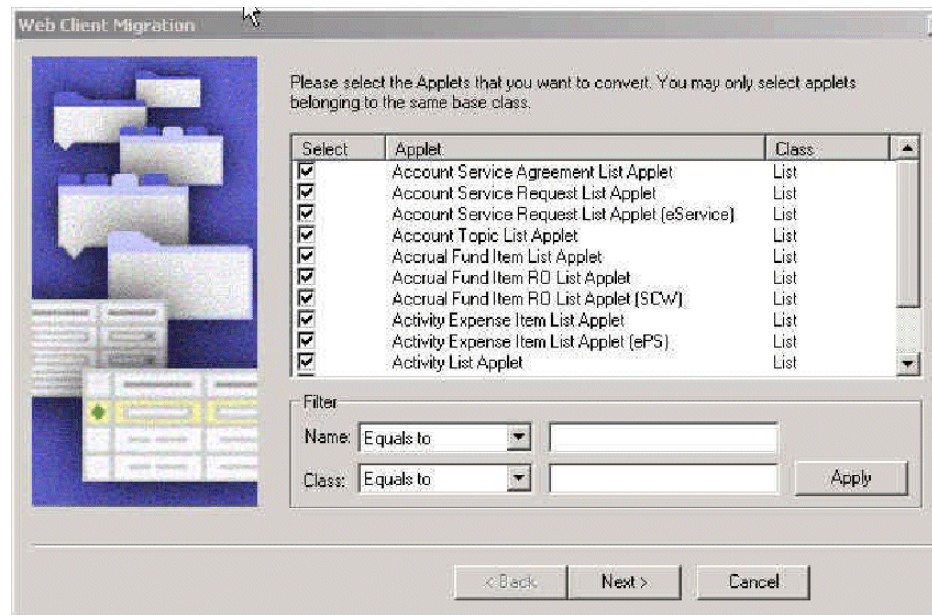
- 1 Before invoking the Web Layout Wizard, identify a model applet that contains the desired template and layout.

NOTE: You may need to modify an applet to obtain the desired characteristics.

- 2 Select the applets you want to remigrate using the selected model applet.

You must select more than one applet.

- 3 Launch the Web Layout Wizard, by right-clicking and selecting Web Layout Wizard.



- 4 Select the applets from the Applet List.

NOTE: You can only migrate one type of applet during each invocation. The wizard will not let you select applets of different types during a single invocation.

- 5 Select the model applet you identified in [Step 1 on page 229](#).

For list applets, you can also specify the maximum number of List Columns that are to be initially visible. The remainder are available from the “Columns Displayed” dialog at run-time.

- 6 Convert the applets.
- 7 Review the conversion results for the set of applets converted in [Step 6](#). If you are satisfied, go to [Step 8](#). Otherwise, return to [Step 6](#).
- 8 Repeat [Step 4 on page 228](#) through [Step 6 on page 230](#) for each Class of applet.

Reviewing Modified Applet Migration

After you completing remigration, you should review the remigrated applets.

To review modified applet migration

- 1 In Siebel Tools, go to the Applet List View.
- 2 Search for applets that are marked as “MigM.”
- 3 Sort by Class.
- 4 For each applet in a given Class, right-click on the applet and select Edit Web Layout.
- 5 Review the layout and make any appropriate changes to the layout using the Applet Web Editor.
- 6 Repeat until you have reviewed all applets.

Reviewing View Migration

After you have migrated views, you should review them to make sure that their Web Layout is accurate.

To review new view migration

- 1 In Siebel Tools, go to the View List view.
- 2 Search for views that are marked as “MigN.”
- 3 For each view of a given type, right-click on the view and select Edit Web Layout.
- 4 Review the layout and make any appropriate changes to the layout using the View Web Editor.
- 5 If you notice that certain changes need to be made for a large number of views, you can remigrate the views using the Web Layout Wizard.

Common problems that may require remigration:

- The mode of a View Web Template Item is incorrect.
- The Template needs to be changed.

NOTE: Changes that do not fall into any of the above categories need to be implemented manually for each view.

Remigrating Views

If you find that a number of views have not been properly migrated, you need to remigrate them using the Web Layout Wizard.

To remigrate views

- 1 Before invoking the Web Layout Wizard for a set of views, identify a model view that contains the desired template and layout.

NOTE: You may need to modify a view to obtain the desired characteristics.

- 2 Select all the views to be remigrated and launch the Web Layout Wizard in Siebel Tools by right-clicking and selecting Web Layout Wizard.
- 3 Select all the views that you wish to remigrate using the selected model view.

NOTE: You can not select views of different types during a single invocation. The wizard will not let you select views of different types during a single invocation.

- 4 Select the model view identified in [Step 1 on page 231](#).
- 5 Convert the views.
- 6 Review the conversion results for the set of views converted in [Step 5](#). If you are satisfied, go to [Step 7](#). Otherwise, return to [Step 1 on page 231](#).
- 7 Repeat [Step 4 on page 231](#) through [Step 5](#) for each type of view.

Reviewing Modified View Migration

After you have remigrated views, you need to review them for accuracy.

To review modified view migration

- 1 In Siebel Tools, go to the Views List view.
- 2 Search for views that are marked as “MigM.”
- 3 For each view, right-click on the view and select Edit Web Layout.
- 4 Review the layout. Make any appropriate changes to the layout using the View Web Editor.
- 5 Repeat until you have reviewed all remigrated views.

Migration of Scripts

The following sections describe in detail how to analyze scripts that need to be modified during the migration, and the process of making these modifications. This section first discusses scripts attached to UI objects—applets and controls. The final section discusses Business Component, Business Service, and Application scripts that reference UI methods.

Controls

In Release 7.x, the `ControlName_Click` event handler has been replaced by the “Method Invoked” property on Control objects. By specifying this property, when a user clicks on a control (for example, a Button), the client framework will check to see if the method has been implemented in Browser Script associated with the `Applet_PreInvokeMethod` event. If the method has not been implemented in Browser Script, the request will be sent to the server for execution. After this request is on the server, the Object Manager will check to see if the method has been implemented in the `WebApplet_PreInvokeMethod` event and then the `BusinessService_PreInvokeMethod` event. If the method has not been implemented in either Browser or Server script, an error will be raised.

In addition to the obsolescence of the `ControlName_Click` event handler, control methods that were formerly available in Siebel VB or eScript are no longer available. In Release 7.x, all control methods are available through Browser Scripting and will execute in the High Interactivity applications only. [Table 23](#) provides a list of the control methods that are supported by Browser Script. For more information, see *Siebel Tools Online Help, MidMarket Edition*.

Table 23. Control Method Syntax Summary

Method	Description	Syntax
Applet	Returns the parent applet for the control.	<pre>var oControl; var oApplet; oApplet = oControl.Applet();</pre>
Buscomp	Returns the corresponding business component for the control.	<pre>var oControl; var busComp; busComp = oControl.Buscomp();</pre>

Table 23. Control Method Syntax Summary

Method	Description	Syntax
GetValue	Returns the value of a control.	<pre>var oControl; var sValue; sValue = oControl.GetValue();</pre>
Name	Returns the name of the control.	<pre>var oControl; var sName; sName = oControl.Name();</pre>
SetValue	Sets the contents of the control to the indicated value.	<pre>var oControl; oControl.SetValue(value);</pre>

In addition to specifying the “Method Invoked” property, controls may also be scripted based upon native DOM events supported by the Browser. [Figure 11](#) depicts the DOM events that may be scripted when a MiniButton control is added to an applet. The scripting of control-based DOM events is supported by High and Standard Interactivity applications and must be implemented in JavaScript.

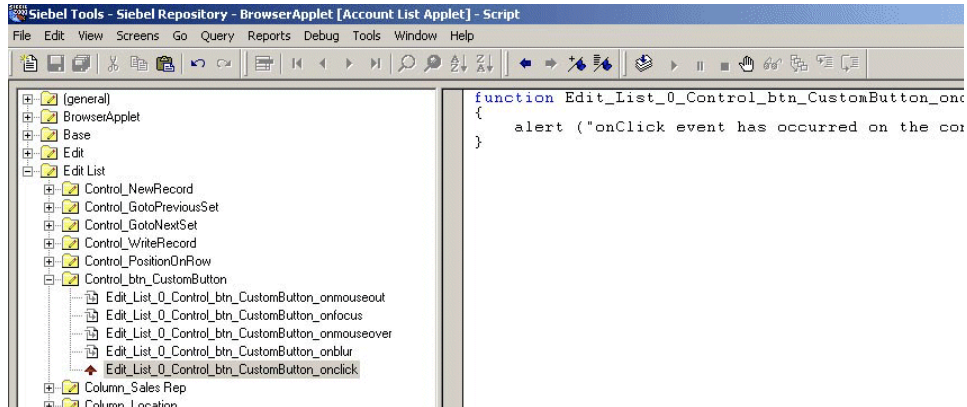


Figure 11. DOM Events When MiniButton Control Added to Applet

Applets

As shown in [Figure 12 on page 236](#), Applet Scripts implemented in Siebel 6.x will need to be moved to one or more of the following Browser or Server Script events:

- (Web) Applet Server Script
 - PreInvokeMethod
 - InvokeMethod
 - ShowControl
 - ShowListColumn
 - PreCanInvoke
 - Load
- Applet Browser Script
 - PreInvokeMethod
 - InvokeMethod
 - ChangeRecord
 - ChangeFieldValue
- Application Server Script
 - PreNavigate
 - Navigate

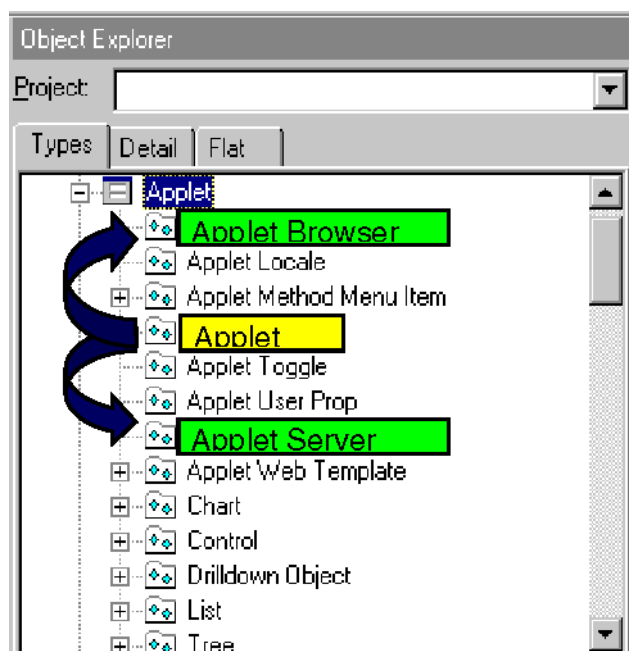


Figure 12. Movement of Applet Scripts

NOTE: In Release 7.x, the WebApplet_ShowControl and WebApplet_ShowListColumn event handlers are only supported for Standard Interactivity applications.

In many cases, the migration of Applet scripts will be relatively straightforward as several WebApplet events correspond to Siebel 6.x Applet events. Please note that though corresponding Browser or Server script event handlers may be available, the scripts will not be automatically migrated. [Table 24](#) depicts Siebel 6.x events, and the corresponding Release 7.x events available in either Browser or Server script.

Table 24. Siebel 6.x Applet Events and Release 7.x Equivalents

Siebel 6.x Applet Event	Release 7.x Equivalent
Applet_Load	WebApplet_Load (Server Event)
Applet_GotFocus	WebApplet_Load (Server Event)
Applet_ChangeFieldValue	Applet_ChangeFieldValue (Browser Event)
Applet_ChangeRecord	Applet_ChangeRecord (Browser Event)
Applet_PreInvokeMethod	WebApplet_PreInvokeMethod (Browser or Server Event)
Applet_InvokeMethod	WebApplet_InvokeMethod (Browser or Server Event)

In cases where there is no direct correlation between Siebel 6.x events and Release 7.x events (for example, Applet_LostFocus event), the functionality of the Siebel 6.x script will need to be evaluated and reimplemented. Depending upon the functionality provided by the script, scripts on the Applet_LostFocus event could be reimplemented as a Server script and implemented on the TheApplication.Navigate or PreNavigate event.

Some applet level methods are no longer available in Release 7.x. The obsolete Applet Methods are listed in [Table 25](#). In some cases, the methods have been reimplemented as Browser Script Methods, as in the case of FindControl and FindActiveX Control. Other methods, such as PostChanges have been replaced by properties that can be configured in Siebel Tools (Immediate Post Changes property on a Field), and a few others will require modifications to scripts that use these methods.

Table 25. Obsolete Applet Methods

Applet Method
ActiveControl
FindControl
FindActiveXControl
GotoControl
PostChanges
Drilldown

Business Components, Business Services, and Application Scripts

The majority of Application, Business Components, and Business Services scripts implemented in Siebel 6.x can remain unchanged and should not require any modification after the upgrade. In instances where existing scripts reference methods that have been made obsolete or contain references to the Siebel user interface, the script will need to be edited. [Table 26](#) lists the Application and Business Component methods that have been made obsolete in Release 7.x.

Table 26. Obsolete Business Components, Business Services, and Application Scripts

Obsolete Business Components, Business Service, and Application Scripts
Application.MsgBox
Application.InputBox

Table 26. Obsolete Business Components, Business Services, and Application Scripts

Obsolete Business Components, Business Service, and Application Scripts
Application.ActiveBusComp
Application.ActiveApplet
Application.GotoApplet
Application.ShowStatus
Application.ActiveControl
Application.FindApplet
BusComp.AllowNotification
BusComp.SuppressNotification

In many cases, implementation alternatives are available in Release 7.x that offer comparable functionality to the obsolete methods. For example, the ActiveBusComp method can typically be replaced with a combination of ActiveBusObject and GetBusComp. By using these two methods together in a script, a handle to the active business component can be obtained. Additionally, MsgBox and InputMox methods can typically be replaced with some core functions available in JavaScript (through Browser script) and some new methods introduced in Release 7.x. Using Browser Script for example, you can use code JavaScript features to provide warnings, dialog boxes or input boxes through the alert, confirm and prompt functions. In order to provide an alert notification from Server Script, the RaiseError and RaiseErrorText methods were introduced (for detailed information on RaiseError and RaiseErrorText please refer to *Siebel Tools Online Help, MidMarket Edition*). These methods allow an error message to be displayed as an alert in the Web Client, though it should be noted that any lines of code that follow the RaiseError or RaiseErrorText methods will not be executed.

Migration of Client-Side Interfaces

This section describes the migration of client-side interfaces.

Outbound COM Interfaces

Client side application integration in the zero-footprint Web client can be accomplished by using two methods provided by Jscript.Net, Microsoft's implementation of the ECMA 262 language. These methods are:

ActiveXObject – Enables and returns a reference to an Automation object

GetObject – Returns a reference to an Automation object from a file

Using the ActiveXObject function for example, you can access properties and methods of Excel, including the Application object and the ActiveSheet.Cells collection as shown in the example below.

```
var ExcelApp;  
var Sheet;  
ExcelApp = new ActiveXObject("Excel.Application");  
Sheet = new ActiveXObject("Excel.Sheet");  
// Make Excel visible  
Sheet.Application.Visible = true;  
// Place some text in the first cell  
Sheet.ActiveSheet.Cells(1,1).Value = "This is row 1 column A";  
// Save the sheet.  
Sheet.SaveAs("C:\\\\TEST.XLS");  
// Close Excel and quit.  
Sheet.Application.Quit();
```

NOTE: The ActiveXObject and GetObject methods can only be used with Internet Explorer 3.0 or greater. For additional information on the ActiveXObject and GetObject functions, please refer to the Microsoft Web site (<http://www.microsoft.com>) and the Jscript.Net documentation.

If you are deploying the Mobile Web Client, you can use native functions provided by Siebel VB or eScript to handle client side application integration, in addition to the ActiveXObject and GetObject functions described above. The native Siebel VB and eScript functions commonly used for client-side application integration are:

- Siebel VB
 - CreateObject (to access a client side Automation object)
 - Declare (to access an external DLL)
- eScript
 - COMCreateObject (to access a client side Automation object)
 - CORBACreateObject (to access a remote CORBA interface)
 - SELib.DynamicLink (to access an external DLL)

For additional information on the Siebel VB and eScript functions listed above, please refer to *Siebel Tools Online Help, MidMarket Edition*.

Inbound COM Interfaces

In Release 7.x, external applications can interact with the Siebel Web Client through the Mobile Web Client Automation Server and Web Client Automation Server. The Mobile Web Client Automation Server is similar in functionality to the COM Automation Server available in Siebel 6.x, though it does not expose applet or control objects.

In order to allow external applications to interact with the zero-footprint Web client, Release 7.x introduces the Web Client Automation Server. The Web Client Automation Server allows external applications to invoke Business Services and manipulate Property Sets from external, COM based applications. The Web Client Automation Server is implemented as a small COM object resident within the browser (IE 5.0 or greater). The Web Client Automation Server is supported with the High Interactivity applications only.

To enable the Web Client Automation Server, in the [SWE] section of your application's configuration file you will need to make sure that the "EnableWebClientAutomation" parameter is set to TRUE. With this parameter set to TRUE, a small ActiveX Control will be downloaded to the desktop and the SiebelHTMLApplication process will be started. This process will terminate when the Siebel Web Client is gracefully terminated. Please note the modification Browser security settings may be required to allow use of the Web Client Automation Server.

Below is an example of a small Microsoft Visual Basic application that uses the Web Client Automation Server and the EAI Siebel Adapter business service to query for a list of Accounts that start with the letter A and displays the result in a message box.

```
Private Sub Command1_Click()  
  
    Dim siebelApp As SiebelHTMLApplication  
  
    Dim inPs As SiebelPropertySet, siebelMsg As SiebelPropertySet  
  
    Dim listOfAcct As SiebelPropertySet, acctPs As  
    SiebelPropertySet  
  
    Dim outputPs As SiebelPropertySet  
  
    Dim svc As SiebelService  
  
    Dim i As Long, j As Long, iPos As Long  
  
    Dim acctList As String  
  
    Dim status As Boolean  
  
    Set siebelApp = GetObject("",  
    "SiebelHTML.SiebelHTMLApplication.1")  
  
    Set inPs = siebelApp.NewPropertySet  
  
    Set siebelMsg = siebelApp.NewPropertySet  
  
    Set listOfAcct = siebelApp.NewPropertySet  
  
    Set acctPs = siebelApp.NewPropertySet  
  
    Set outputPs = siebelApp.NewPropertySet  
  
    Set svc = siebelApp.GetService("EAI Siebel Adapter")
```

` The following lines will construct a property set to query on the Sample Account

```
` Integration Object

siebelMsg.SetType ("SiebelMessage")

status = siebelMsg.SetProperty("IntObjectName", "Sample
Account")

status = siebelMsg.SetProperty("MessageId", "")

status = siebelMsg.SetProperty("Message Type", "")

listOfAcct.SetType ("ListOfSample Account")

acctPs.SetType ("Account")

status = acctPs.SetProperty("Name", "A*")

iPos = listOfAcct.AddChild(acctPs)

iPos = siebelMsg.AddChild(listOfAcct)

iPos = inPs.AddChild(siebelMsg)

` Now that the Integration Object has been constructed, query
for Accounts starting with A

status = svc.InvokeMethod("Query", inPs, outputPs)

If status then

j = 0

i =
outputPs.GetChild(0).GetChild(0).GetChild(0).GetChild(0).GetProp
ertyCount
```

```
Do While j < i
    acctList = acctList &
outputPs.GetChild(0).GetChild(0).GetChild(0). _
    GetChild(j).GetProperty("Name") & Chr(10) & Chr(13)
    j = j + 1
Loop
MsgBox (acctList)
End If
End Sub
```

NOTE: For additional information on the Mobile Web Client Automation Server or the Web Client Automation Server please refer to *Siebel Tools Online Help, MidMarket Edition*.

Production Environment Preupgrade Tasks

7

This chapter describes the tasks which must be performed before you upgrade the Siebel production environment.

If you do not have a development environment, you still need to complete the whole production process; the development environment instructions are supplements to it, not replacements. If you do not have a development environment, you should upgrade a test environment (using a copy of your production database) before you proceed with the upgrade of your production environment. After you have thoroughly tested the application on a test environment, you are ready to proceed with the production environment upgrade. For more information about upgrading without a development environment, see [“Upgrading Without a Development Environment” on page 294](#).

The upgrade process will vary depending on whether you have licensed Siebel Anywhere and will use it to help perform the upgrade. If you are upgrading directly from one of these versions and have licensed Siebel Anywhere, you can use it to automatically upgrade production clients.

CAUTION: If you plan to use Siebel Anywhere to upgrade your production client software environment, you should test the upgrade process first in a test environment. This is to make sure that you are thoroughly familiar with the process before beginning the production upgrade, and to eliminate errors that can affect the ability of mobile users to synchronize.

Before You Begin

If you have not already done so, copy the Upgrade Planning Worksheet, located in [Appendix A, “Upgrade Planning Worksheet,”](#) and fill out the appropriate fields with the following:

- **Siebel Administrator User Name and Password.** This must be a valid RDBMS user name and password, and it must be set up as a Siebel Systems employee. The employee record must have the “Siebel Administrator” responsibility. `SADMIN` is the default administrator user name and password. If this user does not already exist in your database, or does not have “Siebel Administrator” privileges, then you must add this to your database prior to proceeding with the upgrade.
- **Database Tableowner Account User Name and Password.** For DB2 UDB, Microsoft SQL Server, and Oracle, `SIEBEL` is the default tableowner account user name and password for Siebel eBusiness Applications.
- **Siebel Data Table Space.** Only applicable to Oracle, this is the name of the table space on the Oracle server where the Siebel data tables are stored.
- **Siebel Index Table Space.** The name of the table space on the DB2 or Oracle server where the Siebel indexes are stored.
- **Siebel 4-KB Data Table Space.** The name of the table space on the DB2 server where the 4-KB Siebel data tables are stored.
- **Siebel 16-KB Data Table Space.** The name of the table space on the DB2 server where tables reside whose row length equals greater than 4005 bytes, but less than 16,384 bytes.
- **Siebel 32-KB Data Table Space.** The name of the table space on the DB2 server where tables reside whose row length equals up to 32,768 bytes.

NOTE: Siebel table spaces for the DB2 UDB platform should be database managed table spaces (DMS) rather than system managed table spaces (SMS).

Table 27 provides a check list of preupgrade tasks for the production environment.

Table 27. Siebel Production Environment Preupgrade Tasks

Preupgrade Task
<ol style="list-style-type: none"> 1 Fill out your copy of Appendix A, “Upgrade Planning Worksheet” with information specific to your RDBMS platform. 2 Prepare the custrep.dat file. See “Preparing the Custrep.dat File” on page 248. 3 Verify database sort order. See “Verifying Database Sort Order” on page 248. 4 Back up the production database. See “Backing Up the Production Database” on page 251. 5 Save interface table data. See “Saving Interface Table Data” on page 251. 6 Upgrade your RDBMS software. See “Upgrading Your RDBMS Software” on page 253. 7 Verify database server configuration for your RDBMS. See “Verifying Database Server Configuration” on page 254. 8 Preserve custom indexes on tables. See “Preserving Custom Indexes on Tables” on page 278. 9 Disable customized triggers. See “Disabling Customized Triggers” on page 279. 10 Drop customized views. See “Dropping Customized Views” on page 279. 11 Rename the existing production repository from “Siebel Repository” to “Prior Siebel Repository.” See “Renaming Your Existing Production Repository” on page 281. 12 Record dock objects and visibility rules. See “Recording Dock Objects and Visibility Rules” on page 282. 13 Increase database file size. See “Increasing Database File Size” on page 282. 14 Prepare your application data. See “Preparing Application Data for Upgrade” on page 283.

Preparing the Production Environment Database for Upgrade

Before beginning the upgrade of your production environment, make sure that the production database configuration meets the database requirements outlined in the section [“Verifying Database Server Configuration” on page 254](#), and meets the requirements depicted in *Siebel Server Installation Guide for Microsoft Windows, MidMarket Edition*.

Preparing the Custrep.dat File

Navigate to your development environment’s DBSRVR_ROOT\PLATFORM directory. Copy the custrep.dat file created by your development environment upgrade. Place the custrep.dat file in the DBSRVR_ROOT\PLATFORM directory for your production environment. This file will be used by the Siebel Upgrade Wizard to import the New Customer Repository.

Verifying Database Sort Order

Sort order (also called collation sequence) is specified during the initial installation of a database and defines the way in which the database will sort character data. Sort order support depends on both the code page of the database and whether it will be used in a development or a production environment.

- **Development environments.** *You must use a binary sort order* for your development environment database or your repository merges will result in repository corruption.
- **Production environments.** Siebel Systems recommends that your production database use binary sort order (identity sort order for IBM DB2 UDB).

For information about setting the order in which your database sorts data, see *Siebel Server Installation Guide for Microsoft Windows, MidMarket Edition*.

Verifying Sort Order on Microsoft SQL Server

Siebel Systems recommends that you set your sort order to *binary* sort at the instance level at the time of your installation or upgrade of SQL server (the default installation setting is typically dictionary). It is also recommended that each database created under this instance that will be used by the Siebel application is also set to binary sort order. If you wish to set sort order to dictionary case-sensitive or dictionary case-insensitive, set these at the instance and database level (make sure they are the same).

Consult your vendor documentation to verify sort order at the instance and database levels and how to set it appropriately.

CAUTION: Dictionary sort (case-sensitive or case-insensitive) performs 20-30% slower than binary sort.

Sort Order Requirements for MS SQL Production Environment Databases

Siebel eBusiness Applications support the following types of sort order for production environments:

- **Unicode UTF-8**
 - Unicode General
 - Unicode Binary
- **Non-Unicode**
 - Binary

CAUTION: This is not the default sort order; review your vendor documentation for instructions on how to implement binary sort order.

- Dictionary—case-sensitive
- Dictionary—case-insensitive

CAUTION: You should be aware that using dictionary sort order in case-sensitive format can degrade system performance by as much as 20 percent unless it is the default sort order for your database.

Verifying Sort Order on Oracle Client

Siebel Systems supports only binary sort order on your Oracle database, since use of non-binary sort orders will result in performance degradation. Sort order on Oracle is determined by the NLS_SORT parameter on the Oracle Client. To verify that your database was created using binary sort order, see [“Verifying Sort Order on Oracle Client” on page 61](#).

[Table 28](#) illustrates the Oracle sort order supported.

Table 28. Oracle Sort Order Support for Production Environment Upgrades

Database	Environment	Code Page	Sort Order
Oracle 8i Enterprise Server (version 8.1.7.3)	Production	1252, 932	Dictionary (case-sensitive and insensitive)
Oracle 9i Enterprise Server (version 9.0)	Production	UTF-8	Binary

Verifying Sort Order on IBM DB2 UDB

Identity sort order is highly recommended for your DB2 UDB production environment database. To verify that your database was created using identity sort order, see [“Verifying Sort Order on IBM DB2 UDB” on page 62](#).

[Table 29](#) illustrates the DB2 UDB sort order supported.

Table 29. DB2 UDB Sort Order Support for Production Environment Upgrades

Environment	Code Page	Sort Order
Production	UCS-2	Identity (binary)
Production	1252, 932	Identity (binary) or System (dictionary)

Backing Up the Production Database

Perform a full backup of the production database, using the utilities specific to your database platform. This backup protects your production repositories and environment.

It is a recommended practice that you back up your database repository at key stages of the production environment upgrade:

- Before any upgrade activity is started
- After upgrading the Siebel Database Schema
- After upgrading the Custom Database Schema

Saving Interface Table Data

Use the appropriate tools for your RDBMS to export any data in your interface tables that you want to retain. During the upgrade process, your interface tables will be dropped and then recreated. After you have upgraded, you can then import this data using the tools for your RDBMS.

NOTE: During the upgrade, all custom indexes on interface tables are dropped from both logical and physical schema.

Identifying and Resolving Duplicates

Before you upgrade to Release 7.5, you need to resolve any duplicates in your Siebel database. To identify duplicates, run the script, `Find_DUP_S_PARTY_ROW_IDS.sql`, located in the `DBSRVR_ROOT\DatabasePlatform\upgrade` directory.

This SQL script will generate a list of duplicates (if any) that you must resolve prior to beginning the upgrade to Siebel 7.5.

Preparing Mobile and Dedicated Users for the Upgrade

Complete the following actions before beginning the upgrade of your production environment:

- 1 Perform a partial synchronization for mobile users, sending all transactions to the server database.

CAUTION: Mobile users must make no further changes to their local databases until the upgrade has been completed. Any changes made before the upgrade has been completed will be lost when they are reinitialized following the upgrade.

- 2 Verify that mobile clients have synchronized and that all changes have been merged into the server database:
 - a Check that no transaction files remain in the synchronization inbox or outbox for any mobile user. The synchronization inbox and outbox for each user is on the Siebel Server.

Microsoft Windows inboxes will be in the
`C:\SIEBSRVR_ROOT\docking\mobileusername\` directory.

Transaction files are in the format *number.dx*; for example, `00000023.dx`.

- b Log on to a Siebel eBusiness Application, such as Call Center, as the Siebel Administrator. Use the Server Administration - Server Tasks screen to make sure that each Transaction Merger task has successfully completed.
 - c Verify that Workflow Monitor and Workflow Action agents have processed all pending requests. If Workflow Manager has completed successfully, the `S_ESCL_REQ` table should not have any rows.
- 3 To prevent synchronization of mobile clients with the database server, stop or disable all Siebel Remote components on all Siebel Servers, as described in *Siebel Remote Administration Guide, MidMarket Edition* and in *Siebel Server Administration Guide, MidMarket Edition*.
 - 4 Disconnect all thin clients from the database server by stopping the appropriate Application Object Managers, as described in *Siebel Server Administration Guide, MidMarket Edition*.

- 5 Make sure that dedicated clients have disconnected from the database server.

The method you use to do this will depend on your database. For example, with an Oracle RDBMS, you would stop the primary listener. However, all RDBMS types require starting the database in restricted mode. Refer to the documentation that you received from your RDBMS vendor for more information.

Upgrading Your RDBMS Software

The new release of Siebel eBusiness Applications might require you to upgrade your RDBMS server software. Review *Siebel System Requirements and Supported Platforms* to determine if a database server upgrade is required. Refer to the documentation provided by your database vendor for specific instructions on performing the database upgrade.

- If you previously installed DB2 UDB, you must upgrade to the latest DB2 UDB version before upgrading to Release 7.5. Verify that you have updated the database to the current fixpack level as described in *Siebel System Requirements and Supported Platforms*.
- If you are performing an upgrade for DB2 UDB, you must use a DB2 database with 4K, 16K and 32K table spaces defined on it. Otherwise, your upgrade will not complete successfully.
- The `NextSeq` user defined function (UDF) from Release 6.x was replaced by an alternate version for Release 7.x, so your DBA needs to verify that only the current version of the UDF is installed.

The new function definition accepts only one parameter, and the old one is dropped.

Carefully read the relevant chapters of *Siebel Server Installation Guide for Microsoft Windows, MidMarket Edition* for the RDBMS software you plan to configure and install, and for instructions for installing `NextSeq`.

Make sure that the RDBMS upgrade was successful before proceeding with the upgrade of your Siebel database.

NOTE: Be sure to upgrade your client database connectivity software on all upgraded machines to the version specified in *Siebel System Requirements and Supported Platforms*.

Verifying Database Server Configuration

Before you begin your upgrade, you must verify your database configuration. While specific requirements vary by database platform, the consequences of exceeding available resources are the same across platforms: a halted upgrade that requires you to allocate time to adjust the environment and then resume the upgrade.

NOTE: If you wish to change your database platform in conjunction with your upgrade, please contact Siebel Technical Services for assistance.

When you upgrade from earlier versions of Siebel eBusiness Applications to Release 7.x, expect database growth in the range of 30-60% across all platforms. The amount by which it grows depends on a number of factors, such as the size of your database prior to upgrade and the RDBMS software you use.

For estimates of growth percentages for specific RDBMS platforms and releases, see [Table 30 on page 255](#).

NOTE: For all platforms, the growth percentage will increase depending on how you size your database and configure default storage for database table spaces. For example, if you set the default storage for your initial or next extent in a given Oracle table space to 10 KB, that table space will grow by a smaller percentage than if you set it to 100 KB.

Table 30. Siebel Database Expected Growth During Upgrade

Release From Which You Are Upgrading	DB2 UDB	MS SQL Server	Oracle
Release 6.2x	4-KB table space: 200 % 16-KB table space: 400 % 32-KB table space: reduced by 50 %	data 40 %-50 % index 60 %-80 %	tables 40 %-60 % indexes 70 %-80 %
Release 6.0x	4-KB table space: 200 % 16-KB table space: 400 % 32-KB table space: reduced by 50 %	data 40 %-50 % index 60 %-80 %	tables 40 %-60 % indexes 70 %-80 %
Release 5.5x	N/A	data 50 %-60 % index 80 %-100 %	40 %-80 %

NOTE: Actual expected growth may vary widely from these estimates, depending on your database configuration, row size of tables, and data content.

Review the configuration requirements for your specific database platform and make sure that your production database server configuration meets or exceeds them.

- For server configuration requirements for IBM DB2 UDB, see [“DB2 Universal Database Server Configuration” on page 261](#).
- For server configuration requirements for Microsoft SQL Server, see [“Microsoft SQL Server Database Server Configuration” on page 256](#).
- For server configuration requirements for Oracle, see [“Oracle Database Server Configuration” on page 259](#).

Microsoft SQL Server Database Server Configuration

This section contains guidelines for obtaining optimal performance from the Microsoft SQL Server database for use with Siebel eBusiness Applications.

NOTE: These settings should be used only as guidelines for your upgrade configuration. Your final settings will vary based on the server hardware configuration, the number of users, and the type of workload.

Additional information on the configuration of Microsoft SQL Server is available in the Microsoft documentation, information provided by your hardware vendor, and other sources. You should also refer to the Microsoft documentation for additional information concerning tuning options for Microsoft SQL Server.

CAUTION: Never make changes to your Siebel database schema unless instructed on how to do so for a specific purpose by Siebel eBusiness Applications documentation. Otherwise, you may corrupt your entire system and thereby render it unsupportable.

If you are upgrading from Siebel eBusiness Applications version 5.x or 6.x, you must upgrade the Microsoft SQL Server database from 7.0 to SQL Server 2000, using Microsoft's documentation and tools. After you have upgraded SQL Server, be sure that you configure it according to the following recommendations before proceeding with the upgrade.

NOTE: The development database must use binary sort order. Production databases are not constrained by this requirement. This is not the default for MS SQL.

Temporary Database Space

This is the database that Microsoft SQL Server uses for temporary space needed during execution of various queries.

- The default size of `TEMPDB` is too small for almost all production installations. Make `TEMPDB` as big as the biggest table in the Siebel database, or half the size of the Siebel database.
- Make sure that the files used by `TEMPDB` are configured to allow auto-growth. This will allow SQL Server to expand the temporary database as needed to accommodate your activity. Alternatively, you can set `MAXSIZE` to the size of the biggest table or to 50% of the size of the Siebel database.
- It is always beneficial to put `TEMPDB` on a separate drive for performance reasons.

NOTE: Execute `dbcc shrinkdatabase` against `TEMPDB` before starting the upgrade.

Configuration Parameters

[Table 31](#) describes Microsoft SQL Server database configuration parameters that must be set for the upgrade. Before upgrading a MS SQL Server database, make sure that your production database server meets these parameters. For parameters not listed in this table, it is recommended that you accept the default settings.

Table 31. Microsoft SQL Configuration Parameters

Parameter	Setting/Comment
Max. degree of parallelism	0
Cost threshold for parallelism	5
Fill factor (%)	90
Index create memory (KB)	0

NOTE: Most of these parameters do not differ from the default settings.

Siebel Database Options

Set the following Siebel database options to `ON` (enabled) for the upgrade. After your production environment upgrade is complete, you will need to reset these options to their installation settings, as described in [“Production Environment Postupgrade Tasks” on page 319](#).

truncate log on chkpt. This option should be set to `ON` (enabled) for upgrade only. Also, for upgrade only, execute the `alter database` command against Siebel database specifying `set recovery simple` option.

NOTE: You must revert to the original database recovery setting after the upgrade.

turn page detection. Set this option to `ON` (enabled).

auto create statistics. Set this option to `ON` (enabled).

auto update statistics. Set this option to `ON` (enabled).

Rebuild Clustered Indexes

If you have large tables that are extensively used (such as `S_EVT_ACT`, `S_CONTACT`, `S_OPTY`, `S_OPTY_POSTN`, `S_ORG_EXT`), use the MS SQL Server `create index` command with `drop_existing` clause to rebuild large tables with high fillfactor (60%-70%).

Update Statistics

The cost-based optimizer in Microsoft SQL Server uses statistics about tables and indexes to compute the most efficient access plans. When statistics become inaccurate, which can happen for tables with high insertion rates, high deletion rates, or both, and for their associated indexes, the performance of database operations can degrade dramatically.

To update statistics, run `sp_updatestats` to refresh statistical information in the Siebel database.

Oracle Database Server Configuration

Before upgrading an Oracle database, verify the following:

- Upgrade your Oracle database and client software to the version referenced in *Siebel System Requirements and Supported Platforms*, using Oracle's tools and documented procedures.
- Make sure that the system table space has sufficient room for growth, since a number of additional objects are created during the upgrade.

Table Spaces

Allow for growth of the database during the upgrade. Make sure that table spaces have enough free space to accommodate expected growth as estimated in [Table 30 on page 255](#).

pctincrease. For upgrades, you should have a high value `pctincrease` for the table spaces that contain application table spaces and indexes.

pctfree. Before you start the upgrade, rebuild some of your larger tables with a large value of `pctfree` for the larger tables (30 or higher). Examples of large tables are:

- `s_contact`
- `s_evt_act`
- `s_srv_req`
- `s_src`
- `s_org_ext`
- `s_addr_org`

NOTE: The reason that you need to increase `pctfree` before the upgrade is that many new columns are added to these tables in Release 7.x. Migrating data into the new columns during the upgrade is likely to cause row chaining, which will cause performance degradation.

Database Parameters

If you are executing a Siebel Database Upgrade on an Oracle database server with only one CPU, you must change the Oracle Parallel Index setting in the master upgrade ucf file from Y to N. This step must be performed after Siebel Database Configuration and before execution of the Siebel Upgrade Wizard. If this action is not taken, then certain statements that are executed during the upgrade will fail.

If the database server is a multiple-CPU machine, you can take advantage of parallel index creation. Siebel eBusiness Applications assume that parallelism is supported for production environment upgrades; therefore, the Oracle Parallel Index parameter (in the master configuration file) defaults to a value of Y for production environment upgrades. However, if your database server is a single-CPU machine, you need to edit the master configuration file and change the value of the Oracle parallel index parameter to N.

CAUTION: If your database server is a single-CPU machine, you must reset the Oracle parallel index parameter to N, or the operation will fail.

The master configuration file is located in `SIEBSRV/bin`. The file that you need to modify depends on the Siebel version that you are upgrading from, and whether you are upgrading the Siebel database schema (upgrep) or your custom database schema (upgphys). For example, for upgrades of the Siebel database schema (upgrep) in a production environment from Release 6.0.1 to Release 7.x, you need to edit the master configuration file called `master_upgrep_prod_601.ucf`.

The DBA should set appropriate values for the following `init.ora` parameters, depending on the number of CPUs on the database server:

- `parallel_max_servers`

NOTE: The parameter `parallel_max_servers` must be set greater than 1 to enable parallelism.

- `parallel_min_servers`
- `db_block_lru_latches`

The other `init.ora` parameters that the DBA should carefully choose are:

- `sort_area_size`
- `sort_area_retained_size`
- `log_buffers`

For more information, see your Oracle documentation.

Rollback Segments

Rollback segments should be appropriately sized so that the largest of transactions can be accommodated. The upgrade may affect some of the largest tables in your implementation of Release 7.x, causing them to grow by as much as 40%.

The shared pool size should be sufficiently large.

DB2 Universal Database Server Configuration

This section provides guidelines for obtaining optimum performance from a DB2 Universal Database.

Before upgrading a DB2 database, verify that your development database server meets or exceeds the following OLTP parameters:

- You have at least 25 primary and 100 secondary logs of at least 32-MB log size.
- Locklist parameters are set to at least 5,000, and preferably 15,000.
- Maxlocks should be set to 20.
- DMS table space has at least 25% of free pages. If it does not, you will need to increase table space size by adding containers.
- The file system has sufficient space to allow your DMS table space to grow.

In addition, verify that the table spaces are not near their capacity. This can be done by connecting to the database and issuing the following command:

```
db2 list tablespaces show detail
```

Upgrade Instance

After upgrading your RDBMS software (for DB2 UDB, follow the instructions provided by IBM), upgrade your DB2 UDB instance. Important procedures to run are:

To upgrade the instance

- Run the following command on the database server to upgrade to the current version of your RDBMS software:

```
db2updv7 -d DB_NAME
```

Verify DB2 on AIX Permissions

If you are running DB2 UDB on the AIX platform, perform the following steps before executing the Siebel Database Upgrade:

- 1 Navigate to the instance home directory.
- 2 Use the following command to verify that the directory `sqllib/function/routine/sqlproc` has write permission for the group.

```
ls -ld sqllib/function/routine/sqlproc
```

- 3 To authorize group write permission, enter the following command:

```
chmod g+w sqllib/function/routine/sqlproc
```

Increasing the Number of ODBC Statement Handles

DB2 UDB can quickly run out of ODBC statement handles, depending on the number of business objects your enterprise uses. Because it is difficult to know how many business objects your users actually use, you should increase this number automatically each time you install or upgrade the DB2 UDB client, or when rebinding database utilities.

Siebel Systems recommends that you increase the number of CLI packages to six by rebinding the CLI packages, using the special DB2 `CLIPKG` bind option.

To rebind the CLI packages

- 1 Navigate to `C:\sqlllib\bnd` from a DB2 Command window.
- 2 Connect to the DB2 UDB database.
- 3 Enter the following command:

```
db2 bind @db2cli.lst blocking all grant public clipkg 6
```

For more information about the DB2 bind command and the `CLIPKG` option, please refer to IBM DB2 technical documentation.

Upgrade-Specific Parameters

The default settings of the parameters in this section should be adjusted for upgrading to Release 7.5. The values recommended in the following pages are guidelines only and your environment may require adjustments to these, based on numerous factors.

After your upgrade has been completed, and prior to running Release 7.5 in a production environment, you must adjust the DB2 parameters described in this section to those values recommended in *Siebel Server Installation Guide for Microsoft Windows, MidMarket Edition*. Otherwise, your Siebel Database Server may not provide optimal performance.

See your IBM DB2 technical documentation for additional information on DB2 parameters.

DB2 Database Manager Configuration Parameters

You can set the database configuration parameters using the `update database manager configuration` command of the DB2 Command Line Processor or using the DB2 Control Center.

NOTE: Refer to your IBM DB2 UDB technical documentation for more information on modifying the database configuration parameters.

[Table 32](#) provides guidelines for setting DB2 database manager configuration parameters specifically for an optimal upgrade of your Siebel Database. Set these parameters for each DB2 instance. Use the configuration information below for the listed parameters. For parameters not listed in this table, accept the default settings.

Table 32. DB2 Database Manager Configuration Parameters

Parameter	Explanation	Setting
SHEAPTHRES	Sort heap threshold (4 KB)	Double the value allocated for SORTHEAP. See “DB2 Database Manager Configuration Parameters” on page 264.
ASLHEAPSZ	Application support layer heap size (4 KB)	15
MON_HEAP_SZ	Database monitor heap size (4 KB)	128 (minimum)
UDF_MEM_SZ	UDF shared memory set size (4 KB)	256
RQRIOBLK	Max. requester I/O block size (bytes)	65535
QUERY_HEAP_SZ	Query heap size (4 KB)	16384
KEEPDARI	Keep DARI process	YES
QUERY_HEAP_SZ	Query heap size (4 KB)	16384
MAX_COORDAGENTS	Maximum number of coordinating agents	MAXAGENTS

db2set Parameters

Use the `db2set` command to set the parameters (for example, `db2set DB2_RR_TO_RS = YES`) referenced in [Table 33](#). (Under Windows, you would access this through the DB2 Command Line Processor, accessible from the DB2 for Windows 2000 client.)

Table 33. db2set Parameters

Parameter	Explanation	Setting
DB2_RR_TO_RS	Improves DB2 performance with Siebel eBusiness Applications. <i>Set to YES only in production environment servers.</i>	YES
DB2_MMAP_WRITE	Recommended setting only; you should evaluate this setting for your particular configuration and environment.	OFF
DB2_MMAP_READ	Recommended setting only; you should evaluate this setting for your particular configuration and environment.	OFF
DB2_CORRELATED_PREDICATES	When set to ON, the optimizer is able to determine whether predicates in a query are related, which permits DB2 to calculate the filter factor more accurately.	ON
DB2_INDEX_2BYTEVARLEN	This parameter must always be set to ON. Otherwise, you will not be able to create indexes with columns greater than 255 bytes.	ON
DB2_PIPELINED_PLANS	Tells the DB2 optimizer to favor pipeline execution plans; in other words, plans which are left deep and have no temporary result sets.	ON
DB2_INTERESTING_KEYS	Limits the number of execution plans generated by the DB2 optimizer.	ON
DB2_PARALLEL_IO	Useful when using RAID devices. For more information, refer to relevant IBM documentation.	ON
DB2_STRIPED_CONTAINERS	Useful when using RAID devices. For more information, refer to relevant IBM documentation.	ON

NOTE: After changing any of these settings, you must perform a `db2stop/db2start` to implement the changes in your DB2 database.

DB2 Database Configuration Parameters

The database configuration parameters can be set using the `update database configuration` command of the DB2 Command Line Processor or using the DB2 Control Center.

NOTE: See the IBM DB2 technical documentation for more information on modifying the database configuration parameters.

Set the parameters in [Table 34](#) for *each* database within an instance on which you run your Siebel eBusiness Applications. For other parameters of the same type, accept the default settings.

Table 34. DB2 Database Configuration Parameters

Parameter	Explanation	Setting
DFT_DEGREE	Degree of parallelism (1 = turn query parallelism off)	1
DFT_QUERYOPT	Default query optimization class	3
DBHEAP	Database heap (4 KB)	7429
CATALOGCACHE_SZ	Catalog cache size (4 KB)	5558
LOGBUFSZ	Log buffer size (4 KB)	512 (For Windows, set this to 256 .)
UTIL_HEAP_SZ	Utilities heap size (4 KB)	5000
LOCKLIST	Maximum storage for lock list (4 KB)	5000 (The setting should never be smaller than this, but may be increased.)
APP_CTL_HEAP_SZ	Maximum applications control heap size (4 KB)	152 (Recommended size may increase or decrease with the number of users.)
STMTHEAP	SQL statement heap (4 KB)	8192

Table 34. DB2 Database Configuration Parameters

Parameter	Explanation	Setting
SORTHEAP	Sort list heap (4 KB)	20,000 - 40,000 Recommended size; this may increase or decrease depending on the amount of memory in the database server machine and the size of the data. A 20,000 setting allows SORTHEAP to increase up to 80 MB.
APPLHEAPSZ	Default application heap (4 KB)	2500 (Recommended size may increase or decrease with the number of users.)
STAT_HEAP_SZ	Statistics heap size (4 KB)	8000
MAXLOCKS	Percentage of lock lists per application	5
LOCKTIMEOUT	Lock timeout (sec.)	90 to 150
CHNGPGS_THRESH	Changed pages threshold	5
NUM_IOCLEANERS	Number of asynchronous page cleaners	Number of CPUs
INDEXSORT	Index sort flag	YES
SEQDETECT	Sequential detect flag	YES
LOGRETAIN	Sequential or circular log files	NO Setting this parameter to YES means that log files will be archived, and the potential exists for the file system contained the log files to fill up if you do not move or archive the logs.
AVG_APPLS	Average number of active applications	1

Table 34. DB2 Database Configuration Parameters

Parameter	Explanation	Setting
MAXFILOP	Maximum DB files open per application	500
LOGFILSIZ	Log file size (4 KB)	64000
LOGPRIMARY	Number of primary log files	25-50 The value of LOGPRIMARY and LOGSECOND together may not exceed 128.
LOGSECOND	Number of secondary log files	Accept the DB2 UDB default value; increase this value if secondary log files are required for your deployment.
SOFTMAX	Triggers buffer pool flushing	50
NUM_IOSERVERS	Number of disks on which the database resides	Number of disks

Installing the Stored Procedures and User-Defined Functions

NOTE: Skip this procedure if you are performing an upgrade from Release 7.0.x to Release 7.5. You already completed this step during your upgrade to Release 7.0.x.

If you are upgrading from Release 5.x or 6.x and your RDBMS is DB2 UDB, you need to install the stored procedures and user-defined functions (UDFs) on the database server. To install the stored procedures and user-defined functions on your database server, you must first transfer them to the database server, and have installed the database server components. (For information on installing database server components, refer to the chapter on installing the Siebel Database Server for DB2 Universal Database in *Siebel Server Installation Guide for Microsoft Windows, MidMarket Edition*.)

If you use a different RDBMS platform, skip to [“Preparing Application Data for Upgrade” on page 283](#).

The user-defined functions (UDFs) and stored procedures must be transferred to and installed on the database server to support the Siebel product. Any method that transfers the necessary files to the correct location on the database server is acceptable.

NOTE: Compiling stored procedures in DB2 creates .dll files. You must copy these files to the database server machine's `sqllib\function` directory.

To copy and install the stored procedure code, follow the procedure appropriate to your database server platform. You first must delete the old stored procedures. Then you need to install the Siebel stored procedure library on the DB2 database server host.

Before you perform this procedure, make sure that you have followed IBM documentation regarding upgrading your DB2 database software to version 7. As part of your DB2 upgrade, you must apply the DB2 fix pack that comes with Siebel software. For information on how to perform basic DB2 tasks, refer to the relevant IBM documentation.

To delete the old stored procedures

- Delete the old stored procedures from the `sqllib/function` directory:
 - On the Windows Database Server:

<code>nextseq.dll</code>	<code>mapping.dll</code>
<code>nextseqd.exe</code>	<code>updt2col.dll</code>
<code>siebstat.dll</code>	<code>updt_addr_org_sp.dll</code>
<code>siebsdmy.dll</code>	<code>updt_addrorg_intsp.dll</code>
<code>siebtrun.dll</code>	<code>updt_addr_per_sp.dll</code>
<code>siebfkst.dll</code>	

- On the UNIX Database Server:

<code>nextseq</code>	<code>updt2col</code>
<code>nextseqd</code>	<code>siebtrun</code>
<code>siebstat</code>	<code>siebfkst</code>
<code>siebupgl</code>	<code>updt_addr_org_sp</code>
<code>siebsdmy</code>	<code>updt_addr_per_sp</code>
<code>mapping</code>	<code>updt_addrorg_intsp</code>

To copy and install the stored procedure code

Install the Siebel stored procedure library on the DB2 database server host.

- 1 Log onto the Siebel source installation machine, and navigate to the source installation subdirectory that contains the Siebel Database installation objects.

The directory that contains the file to install (`siebproc`) is

`DBSRVR_ROOT \DB2UDB\SIEBPROC\DBSRVR_OS`

where:

`DBSRVR_OS` = the operating system your database server runs on; for example, WINNT, WIN32, or aix.

- 2 Put the `siebproc` file (on Windows this is called `siebproc.dll`) into the `FUNCTION` subdirectory within the DB2 UDB instance directory (where DB2 UDB is installed) on the Siebel Database Server.

For example, on Windows, this location might be `C:\SQLLIB\FUNCTION`.

Preparing the Database for a DB2 Upgrade

You need to perform the following procedures to prepare your database for a DB2 UDB upgrade:

If you are upgrading from Release...	Perform the following procedure...
5.x, 6.x, or 7.0.x	Create DB2 temporary table spaces and buffer pools.
5.x, 6.x, or 7.0.x	Increase 4-KB table spaces.
5.x	Create 16-KB table spaces and buffer pools.
6.x or 7.0.x	Increase 16-KB table spaces.
5.x or 6.x	Create 32-KB table spaces and buffer pools.
7.0.x	Increase 32-KB table spaces and buffer pools.
5.x, 6.x, or 7.0.x	Determine table space page size requirements.
5.x, 6.x, or 7.0.x	Verify that the Application Development Client/Tool is installed on your database server.

Creating DB2 Temporary Table Spaces and Buffer Pools

If your RDBMS is DB2 UDB, verify that you have 16-KB and 32-KB temporary table spaces to use for sorting and other SQL processing. Both the 16-KB and 32-KB temporary table spaces require dedicated buffer pools.

To create a 16-KB temporary table space

- 1 Create a 16-KB buffer pool with at least 5,000 16-KB pages.
- 2 Create a 16-KB temporary table space as system managed space (SMS) that can be expanded to 2 GB of storage.

To create a 32-KB temporary table space

- 1 Create a 32-KB buffer pool with at least 1,000 32-KB pages.
- 2 Create a 32-KB temporary table space as SMS that can be expanded to 2 GB of storage.

Increasing DB2 4-KB Table Space

If you are upgrading from Release 5.x, 6.x, or 7.0.x and your RDBMS is DB2 UDB, you must increase the size of your 4-KB table space.

To increase the size of your 4-KB table space

- Increase the 4-KB table space using the DB2 tools of your choice. For sizing requirements, see [Table 30 on page 255](#).

This completes the table space allocation for your 4-KB table space.

Creating DB2 16-KB Table Space and Buffer Pool

If you are upgrading from Release 5.x and your RDBMS is DB2 UDB, create a new 16-KB table space and buffer pool, to which the data in your previous installation's 8-KB table space will be migrated.

NOTE: Skip this step if you are upgrading from Release 6.x or 7.0.x, because you already created your 16-KB table space and buffer pool. Instead, perform the task described in [“Increasing DB2 16-KB Table Space” on page 273](#).

To create 16-KB table space and buffer pool

- 1 Create a 16-KB buffer pool, with a number of pages equal to at least twice the number of pages in the 8-KB buffer pool in your previous installation.

NOTE: If your system has limited RAM, you may want to reduce the size of your current 8-KB buffer pool or increase the size of your swap file before upgrading; this will prevent an upgrade failure because of lack of memory.

- 2 Create a 16-KB table space using the DB2 tools of your choice.

This completes creation of your new 16-KB table space and buffer pool.

Increasing DB2 16-KB Table Space

If you are upgrading from Release 6.x or 7.0.x and your RDBMS is DB2 UDB, you must increase the size of your 16-KB table space.

NOTE: Skip this step if you are upgrading from Release 5.x.

To increase the size of your 16-KB table space

- Increase the 16-KB table space using the DB2 tools of your choice. For sizing requirements, see [Table 30 on page 255](#).

This completes the table space allocation for your 16-KB table space.

Creating DB2 32-KB Table Space and Buffer Pool

If you are upgrading from Release 5.x or 6.x and your RDBMS is DB2 UDB, you must create a new 32-KB table space and buffer pool in order for the upgrade to complete successfully.

NOTE: Skip this step if you are upgrading from Release 7.0.x, because you already created your 32-KB table space and buffer pool. Instead, perform the task described in [“Increasing DB2 32-KB Table Space” on page 274](#).

To create 32-KB table space and buffer pool

- 1** Create a 32-KB buffer pool, with the number of pages equal to at least 1,000 32-KB pages.
- 2** Create a 32-KB table space using the DB2 tools of your choice.

This completes creation of your new 32-KB table space and buffer pool.

Increasing DB2 32-KB Table Space

If you are upgrading from Release 7.0.x and your RDBMS is DB2 UDB, you must increase the size of your 32-KB table space.

NOTE: Skip this step if you are upgrading from Release 5.x or 6.x.

To increase the size of your 32-KB table space

- Increase the 32-KB table space using the DB2 tools of your choice. For sizing requirements, see [Table 30 on page 255](#).

This completes the table space allocation for your 32-KB table space.

Determining Table Space Page Size Requirements for DB2 UDB

There are four standard database managed table spaces (DMS) which hold Siebel tables and indexes—a 4-KB, 16-KB, 32-KB table space, for various sized tables, and a table space to hold indexes. However, in some cases, you may have Siebel tables in custom table spaces.

If a custom table has an estimated page size greater than its current table space page size, it will not fit in its table space after the upgrade, and the upgrade will fail. The following utility will determine whether each of your current custom tables will increase in size to the point that it will require a move to a larger table space.

This utility must be run prior to the upgrade, before any steps of the upgrade are executed.

To determine table space page size requirements for DB2 UDB

- 1 From the `siebsrvr/bin` directory, type the following command line:

```
tblsize /U $Tableowner /P $Password /C $ODBCDataSource /F
$DDLFilename /B $DefaultTablespace /X $DefaultIndexspace /K
$16Ktablespace /V 32Ktablespace /Q $ReportFilename /L
$LogFilename
```

where:

- `Tableowner` = Tableowner
- `Password` = Tableowner password

- `ODBCDataSource` = Data source of the database
 - `DDLFilename` = Name of the DDL file (This file is called `ddl.ct1`, and it is located in the `dbsrvr/db2udb` directory.)
 - `DefaultTablespace` = Name of the 4-K page standard Siebel table space
 - `DefaultIndexspace` = Name of the standard Siebel indexspace
 - `16Ktablespace` = Name of the 16-K page standard Siebel table space
 - `32Ktablespace` = Name of the 32-K page standard Siebel table space
 - `ReportFilename` = Name of the report generated by the utility
 - `LogFilename` = Name of the log file (The default name is `custtbl1.log`.)
- 2** Review the report generated by the utility to determine if the status of the estimated table pagesize postupgrade is larger than the size of the custom table pagesize.

An example of the report generated by this utility is provided below:

```
Table Name = S_EVT_ACT
Custom Tablespace Id = 5
Custom Tablespace Name = CUST_TBS_EVT_ACT
Custom Tablespace Pagesize = 4096
Estimated Table Pagesize (Postupgrade) = 5067
Status = Does not fit in its custom tablespace
```

- 3** For each table which has Status: Does not fit in its custom tablespace, you must create a larger custom table space which is larger than the estimated table pagesize postupgrade.

- 4 Move the tables from their old table spaces to their new table spaces by running `ddlmove`.

DDLMOVE is a utility for moving tables from one table space to another table space. This utility is located under the `siebsrvr\BIN` directory.

To run `ddlmove`, submit the following arguments:

```
ddlmove /U $Tableowner /P $TablePassword /C $ODBCDataSource /E
$Stop_on_DDL_Error /G $Grantee /B $Tablespace /X $IndexTablespace
/M $TableName /L $LogFilename /Z $UCS2Database
```

where:

- `Tableowner` = Table owner of the database (Required)
- `TablePassword` = Password of the table owner of the database (Required)
- `ODBCDataSource` = Data source of the database
(Default environment variable: `SIEBEL_DATA_SOURCE`)
- `Stop_on_DDL_Error` = Stop on DDL Error (Default: Y)
- `Grantee` = Grantee for tables
- `Tablespace` = Name of the table space that you are moving the table to
- `IndexTablespace` = Name of the index space that you are moving the table to
- `TableName` = Table Name Like Support (Default: N)
- `LogFilename` = Name of the log file (The default name is `ddlmove.log`.)
- `UCS2Database` = (Default: N)
- `TableGroupingsFilename` = Name of the table grouping file

NOTE: If there are problems reported by the sizing utility, you must resolve the table space page sizes before you proceed with the upgrade.

Verifying Installation of the DB2 UDB Application Development Client

If you are upgrading to Release 7.5 from Release 5.x, 6.x, or 7.0.x and your RDBMS is DB2 UDB, verify that the DB2 UDB Application Development Client is installed on your database server before you proceed with the upgrade. To do this, navigate to the appropriate directory on the database server and verify that the DB2 UDB Application Development Client is installed.

Table 35 lists the DB2 UDB Application Development Client components that must be installed on your database server depending on your platform. Use this table to verify that you have the correct components installed on your database server.

Table 35. DB2 UDB Application Development Client Components

Platform	DB2 UDB Application Development Client Components
Windows	DB2 Application Development Client
AIX	db2_07_01.adt.rte 7.1.0.xx COMMITTED Application Development Tools db2_07_01.adt.samples 7.1.0.xx COMMITTED ADT Sample Programs
HP	DB2V7SKL 7.1.0.40 Application Development Tools for HP-UX
Solaris	application db2adt71 Application Development Tools (ADT) (PTF 1720500-00) application db2adts71 ADT Sample Programs (PTF 1720500-001)

If the DB2 Application Development Client is not installed, you must install it. For more information, refer to the relevant IBM documentation.

CAUTION: On the Windows platform, client installation directory paths are limited to 18 characters. If your previous installation directory name was more than 18 characters (for example, C:\YOURCOMPANYNAME\siebel, where *YOURCOMPANYNAME* is more than 12 characters long), you need to choose a new installation directory. Support for specifying a new installation directory is provided by SiebelAnywhere.

CAUTION: Make sure that attachment files for obsolete tables are renamed or copied to alternate locations.

For example, since S_EMPLOYEE_ATT is migrated to S_CONTACT_ATT, you need to rename a file such as S_EMPLOYEE_12-1ABC.SAF to S_CONTACT_12-1ABC.SAF.

Preparing Tables for Upgrade

Take the following measures to prepare your tables for the upgrade.

Preserving Custom Indexes on Tables

Release 7.x upgrade drops and recreates custom indexes on base tables. However, custom indexes on interface tables are not recreated during the upgrade, but they can be restored when the upgrade is complete.

NOTE: Custom indexes may need to be changed to reflect schema changes. You should reevaluate custom indexes for applicability in the new release.

For more information about applying custom indexes, see *Siebel Tools Reference, MidMarket Edition*.

CAUTION: If your DBA created custom indexes in your current schema that were not defined through Siebel Tools in the Siebel repository, then custom indexes will be dropped during the upgrade process.

Considerations for Clustered Indexes

If you created clustered indexes on base tables, and Release 7.x introduces a different clustered index on the same table, the upgrade process will recreate custom indexes as nonclustered and create the Siebel index as clustered.

For IBM DB2 UDB, indexes that reside on tables that are defined with append mode enabled will be recreated as nonclustered indexes during the upgrade.

(Tables cannot contain clustered indexes and have append mode enabled.)

Disabling Customized Triggers

Release 7.x does not support customized triggers. If you created customized triggers on your Siebel base tables, please disable them before you perform the upgrade. You will then need to recreate them after the upgrade is finished.

Dropping Customized Views

If you created customized views on your Siebel base tables, you must drop them before you perform the upgrade. If they are still applicable after the upgrade, you will need to recreate them after the upgrade is finished.

Identifying DB2 UDB Long Columns for Truncation

In Release 7.5, the maximum length for DB2 UDB long columns with a type of varchar has reduced to 16350 from 16383. Long columns of type varchar that exceed 16,350 will be truncated.

To prevent a data truncation error that may cause transaction processing (`txnproc`) or transaction routing (`txnroute`) to fail, perform the following steps to identify these columns and reduce the data in these columns.

To identify and reduce the length of long varchar columns

- 1 From any shell, open the script `chk16350.bat`, and edit the following parameters as appropriate for your deployment:

`SRC_USR` = username of the source database

`SRC_PSWD` = password for the source database

`SRC_TBLO` = table owner of the source database

`SRC_TBLO_PSWD` = table owner password for the source database

`SRC_ODBC` = ODBC data source name of the source database (edit the value “CHANGE_ME”)

`SRC_REPOSITORY_NAME` = repository name of the source database

`DBSRVR_ROOT` = directory where you installed the dbsrvr component of the Siebel Server (edit the value “CHANGE_ME”)

`SIEBEL_ROOT` = directory where you installed the siebsrvr component of the Siebel Server (edit the value “CHANGE_ME”)

`VALID_RESULTS_DIR` = directory where you want the output files to be generated (edit the value “CHANGE_ME”); this must be an existing directory

This script produces two files:

- **long_trunc_cols.rpt.** This report identifies all long varchar columns that are longer than 16350 characters.
 - **update_trunc.sql.** This SQL file will generate update statements that truncate identified columns to 16350 characters.
- 2 Reduce the data in these columns using either of the following methods:
 - Manually review the columns in the `long_trunc_cols.rpt` report and manually reduce the size of each column identified.
 - Run `update_trunc.sql` using the DB2 command line processor.

CAUTION: If you do not truncate or otherwise reduce the data in these columns, you will receive a “data truncated” error, and transaction processing and transaction routing may fail.

Preparing the Siebel Repositories for Upgrade

Two separate repositories are used during the production upgrade process:

- **Your existing production repository.** To prevent a naming conflict, before you run the upgrade, rename your existing production repository (*Siebel Repository*) to *Prior Customer Repository*. After the upgrade, your new Release 7.5 production repository will be given the name *Siebel Repository*.
- **New Customer Repository.** This repository will be loaded when you run the upgrade wizard.

Renaming Your Existing Production Repository

Use the version of Siebel Tools that matches your old version of Siebel eBusiness Applications (for example, Siebel Tools Release 6.x for upgrades from Release 6.x). Connect directly to the production database server, and then perform the following steps to make sure that your existing repositories follow the correct naming conventions:

To rename the repository

- 1** Using the appropriate prior version of Siebel Tools, connect to the Siebel Database Server.
- 2** Change the name of the existing repository, as described below:
 - a** In the Object Explorer, click the Types tab.
 - b** Click Repository.
 - c** In the Repository view, click Name.
 - d** Locate the appropriate repository in the list applet and rename it *Prior Customer Repository*.
- 3** Step off the list to commit the record to the database.

NOTE: The upgrade process will verify the repository names. If no repository is named *Prior Customer Repository*, the *Siebel Repository* will be renamed to *Prior Customer Repository* in the target database so that the upgrade will execute properly.

If you need more information about renaming repositories, refer to *Siebel Tools Reference, MidMarket Edition*.

Recording Dock Objects and Visibility Rules

NOTE: This information only applies to deployments without a development environment. If you have a development environment, then you already performed this procedure and you do not need to repeat it.

Changes to visibility rules and dock objects require the assistance of Expert Services.

Modified visibility rules will be dropped during the upgrade. Manually record your changes to dock object visibility rules so you can evaluate the need to reapply the changes after the upgrade is complete.

Dock objects and visibility rules created by using Docking Wizard will be preserved unless they become invalid after the upgrade. Manually record any changes that you made through the Docking Wizard so that you can evaluate the need to reapply the changes after the upgrade is complete.

Increasing Database File Size

If your RDBMS is MS SQL Server, you should increase your database file size by resetting the `Autogrowth` parameter to between 25% and 50%. Failure to do this could diminish upgrade performance and possibly impact the success of your upgrade.

NOTE: Failure to increase your database file size might result in an entry in the `ddctl` log files.

Preparing Application Data for Upgrade

If you use one of the following Siebel eBusiness Applications, you need to perform preupgrade procedures to prepare your data for upgrade to Release 7.x:

- Siebel eChannel
- Quotes
- Timesheet

Siebel eChannel

NOTE: If you are upgrading from Release 7.0.x to Release 7.5, you do not need to perform this procedure.

Before upgrading to Release 7.x, Siebel eChannel customers who have modeled their partners as both Accounts and Divisions (or Organizations) should merge these two records to make sure that only one record will exist for each partner company in the new single party model. Similarly, partner employees that have been modeled as Contacts and Employees should be merged to result in a one-person record in the upgraded database. There is a tool that will help customers find and populate a temporary table with the matching partner company and employee records. Records that have been identified as matches will be merged during the upgrade, creating a single record for each partner company or partner employee. For information about this tool, contact Siebel Professional Services.

Quotes

NOTE: If you are upgrading from Release 7.0.x to Release 7.5, you do not need to perform this procedure.

To make sure quotes that use products as solutions in Release 6.x will be upgraded to Release 7.x, verify that the Model Product field is unchecked.

In Siebel Quotes 6.x, the Discount Amount field on a line item was automatically populated to \$0.00. This meant that, for any line item, users had to clear the discount amount field if they wanted to apply a header level discount after adding a line item.

However, some customers implemented Siebel Quotes 6.x to read the Discount Amount field the same whether it was NULL or \$0.00, thereby allowing users to enter a header discount after adding a line item and avoiding the situation illustrated above.

In Release 7.x, this issue is no longer applicable. The Discount Amount field is left as NULL unless a user explicitly enters a value, including \$0.00. Therefore, when a customer upgrades from Release 6.x to Release 7.x, the customer will need to determine whether no change had been made in Siebel Quotes 6.x to Discount Amount (thus letting it be automatically populated to \$0.00) or whether Siebel Quotes 6.x had been modified to read NULL or \$0.00 in Discount Amount as the same. In the event the latter is true, the customer will need to run a Quote script with the following SQL statement applied before beginning the upgrade, that is, before running the configuration utility and applying it to development and production upgrades.

```
update S_QUOTE_ITEM  
  
set  DISCNT_AMT = NULL  
  
where DISCNT_AMT = 0
```

This script will account for the fact that the Discount Amount field is read the same whether it is NULL or \$0.00. Administrators running Siebel Quotes 6.x without modifications to Discount Amount can ignore this script.

Timesheet

If you are upgrading from a pre-7.5 version of Siebel eBusiness Applications, see Siebel SupportWeb for a technical note titled *Time Sheet Conversion Overview*.

Calendar

NOTE: If you are upgrading from Release 7.0.x to Release 7.5, you do not need to perform this procedure.

An optional preupgrade script may be run to prevent duplicate Siebel Sync appointments in Release 7.x calendar. The consequences of running and not running this script are listed below.

If you run the preupgrade script:

- Users will not see duplicate Siebel Sync appointments in Release 7.x.
- Planned Start value will be modified for any record where Release 6.x Planned Start = Creation AND Start Time IS NOT NULL.

If you do not run the preupgrade script:

- Users will see duplicate Siebel Sync appointments in Release 7.x. This will be particularly problematic for repeating appointments.
- Planned start will not be overwritten.

If you wish to run this script, the following SQL statement must be run before the upgrade (this step cannot be performed after the upgrade):

```
update S_EVT_ACT
set TODO_PLAN_START_DT = null
where APPT_START_DT is not null
and APPT_START_TM is not null
and TODO_PLAN_START_DT = CREATED
```

This step will update Siebel Sync imported records TODO_PLAN_START_DT to null.

Production Environment Preupgrade Tasks

Preparing Application Data for Upgrade

Upgrading the Production Environment

8

This chapter describes the tasks involved in upgrading the Siebel production environment. Do not proceed unless you have completed the preupgrade tasks in [Chapter 7, “Production Environment Preupgrade Tasks,”](#) to prepare your production environment data and your repositories for the upgrade.

After you have completed all of the pre-upgrade tasks outlined in the previous chapter, you are ready to perform the actual upgrade of the production environment. A checklist of upgrade tasks is provided in [Table 36](#).

CAUTION: Be sure to uninstall the previous versions of Siebel server and client software before installing the new versions. This should be done to validate proper functioning.

Table 36. Siebel Production Environment Upgrade Tasks

Upgrade Tasks	
1	Upgrade the Siebel Gateway Name Server and Siebel Servers. See “Upgrading Gateway Name Server and Siebel Servers” on page 289 .
2	Install Siebel database server software. See “Installing the Siebel Database Server Software” on page 296 .
3	Configure Siebel database server software. See “Configuring the Siebel Database Server for Upgrade” on page 297 .
4	Upgrade Siebel database schema. See “Upgrading the Siebel Database Schema” on page 305 .
5	Review Siebel database schema upgrade log files. See “Reviewing the Upgrade Log Files” on page 306 .
6	Troubleshoot the Siebel database schema upgrade. See “Restarting the Upgrade” on page 307 .
7	If necessary, restart the upgrade. See “Restarting the Upgrade” on page 307 .
8	Upgrade the custom database schema. See “Upgrading the Custom Database Schema” on page 309 .
9	Troubleshoot the custom database schema upgrade. See “Troubleshooting the Custom Database Schema Upgrade” on page 316 .
10	Review custom database schema upgrade log files. See “Reviewing the Upgrade Log Files” on page 316 .
11	Add new license keys. See “Adding New License Keys” on page 318 .

Upgrading Gateway Name Server and Siebel Servers

You must upgrade all of your production environment servers—Gateway Name Server and Siebel Servers—to the Release 7.5 software. If you have multiple servers in your production environment, you will need to perform the upgrade procedure described in this section for all Siebel Servers in your deployment.

Before proceeding, make a copy of your previously completed Upgrade Planning Worksheet. You will need to refer to this during the upgrade process for server names and other installation information. You will also need to refer to *Siebel Server Installation Guide for Microsoft Windows, MidMarket Edition* for server installation instructions.

CAUTION: Do not attempt to install the new version of Siebel eBusiness Applications software without first removing the previous version as instructed. If you attempt this, your installation might not operate correctly.

To upgrade your production environment servers, you must perform the following procedures on each server:

- Stop all Siebel Servers and the Gateway Name Server service.
- Uninstall the previous version of the Siebel Server software.
- Uninstall the earlier version of the Gateway Name Server.
- Install the version 7.5 Gateway Name Server software.
- Install the version 7.5 Siebel Server on all application servers.

Upgrading the Servers on the Windows Platform

To stop all servers

- Stop all Siebel Servers and Siebel Gateway Name Server service by navigating to Start > Settings > Control Panel > Services.

To uninstall the earlier version of the production Siebel Servers

- 1** If you wish to preserve your installation-specific Siebel Server configuration parameters, record your current configuration.

You will need to reapply that configuration manually after completing the upgrade. Installation-specific parameters will be lost when you uninstall your existing Siebel Servers. When you install the new Siebel Server Release 7.5 software, you can reset these parameters manually through the Server Manager.

- 2** Uninstall the previous release of the Siebel Server software.
 - For upgrades from Release 5.x, double-click the UnInstall icon in the Siebel Server program folder.
 - For upgrades from Release 6.x, navigate to the Control Panel, and double-click Add/Remove Programs. Select the previous release of Siebel Server, and then click Add/Remove. Follow the instructions in the InstallShield wizard to uninstall the applications.
- 3** Delete the entire Siebel Server installation directory tree, and then restart the machine.

NOTE: To avoid conflicts with the ODBC drivers used by the new Siebel Server software, remove any older versions of the ODBC drivers. If the vendor provided an uninstall option, use it. If the vendor has not provided an uninstall option, contact the vendor for removal instructions.

To uninstall the earlier version of the Siebel Gateway Name Server

- 1** Uninstall Siebel Gateway Name Server:
 - For upgrades from Release 5.x, double-click the UnInstall icon in the Siebel Gateway Name Server program folder.
 - For upgrades from Release 6.x, navigate to Start > Settings > Control Panel > Add/Remove Programs. Select the previous release of Siebel Gateway Name Server, then click Add/Remove. Follow the instructions in the InstallShield wizard to uninstall the applications.
- 2** Delete the entire Siebel Gateway Name Server installation directory tree, and then restart the machine.

To install the version 7.5 Siebel Gateway Name Server software

- Refer to the appropriate chapter on installing the Siebel Gateway Name Server in *Siebel Server Installation Guide for Microsoft Windows, MidMarket Edition*.

To install the version 7.5 production environment Siebel Servers

- 1** Make sure that each application server on which a Siebel Server will be installed has the correct versions of all required third-party software products, as documented in the Release 7.5 *Siebel System Requirements and Supported Platforms*.
- 2** Install the new Siebel Server on all application servers, as described in *Siebel Server Installation Guide for Microsoft Windows, MidMarket Edition*, and the Siebel Database Server.

NOTE: With the Siebel Enterprise Server (SES) installer, when the Siebel Server is chosen, make sure that you choose the Siebel Database Server as well.

NOTE: Refer to your previously completed copy of the Upgrade Planning Worksheet for server names and other installation information.

Installing the Siebel Database Server Software

You will need to install the Release 7.x Siebel Database Server software onto one production Siebel Server. The Siebel Database Server must be installed on a Siebel Server that has already been upgraded to Release 7.x.

NOTE: With the Siebel Enterprise Server (SES) installer, when the Siebel Server is chosen, make sure that you choose the Siebel Database Server as well.

CAUTION: In order to configure and execute Siebel Database Server procedures and maintenance scripts, you must have sufficient access to Siebel Server 7.5 directories:

You must have READ-WRITE access to `BIN` directories under Siebel Server executables in the `SIEBSRVR_ROOT` directory.

You must have READ-WRITE access to the log directories and upgrade directory.

To install the Release 7.x Siebel Database Server software in your production environment, follow the instructions for installing database server components in *Siebel Server Installation Guide for Microsoft Windows, MidMarket Edition*, in the chapter on installing the Siebel Database Server software on your RDBMS. Refer to your completed Upgrade Planning Worksheet for server names and other installation information.

NOTE: Repeat the process outlined in the *Siebel Server Installation Guide, MidMarket Edition* for installing the stored procedures and user-defined functions.

In order to have Siebel Administrator permissions, you must log in with a valid RDBMS user name and password, and this user name must have “Siebel Administrator” responsibility for the default organization.

The default administrator user name is:

SADMIN

If the user `SADMIN` does not exist or does not have “Siebel Administrator” responsibility for the default organization, you must contact your database administrator to establish this before you proceed with the upgrade.

This attribute can be changed to the login of another “employee” if necessary to meet your business needs.

Upgrading Without a Development Environment

If you are upgrading without a development environment, you will follow a slightly different process to upgrade your environment. This section describes this process. During your upgrade, you will first execute all steps against a test environment. After you have verified that the steps work correctly in the test environment, you will execute the steps against your production environment.

Complete the following steps to upgrade your production repository and database schema.

CAUTION: Do not attempt to upgrade your system using these instructions if you have a development environment, and have done any customization or made any configuration changes to your Siebel deployment. If you do, all of your customization will be overwritten during the upgrade.

To upgrade without a development environment

- 1** Upgrade your test environment. Navigate to `/dbsrvr/common` and locate the `mstrep.dat` file. Copy the `mstrep.dat` file and rename it `custrep.dat`. Place the `custrep.dat` file in the `/dbsrvr/platform` directory.
- 2** Use the configuration utility to upgrade the Siebel database schema on your test environment, as instructed in [“Upgrading the Siebel Database Schema” on page 305](#).
 - a** Specify parameters you listed in the Upgrade Planning Worksheet.
 - b** Set the following parameter as shown:

```
Environment Type = "Production"
```

NOTE: You may need to change the configuration of the ODBC data source when you upgrade the Siebel database schema and the custom database schema to point to your test database before executing these scripts.

- 3** Use the configuration utility to upgrade the Siebel custom database schema on your test environment, as instructed in [“Upgrading the Custom Database Schema” on page 309](#).
 - Specify parameters you listed in the Upgrade Planning Worksheet.

- 4 Review the upgrade log files to verify that no unacceptable errors are reported. See [“Reviewing the Upgrade Log Files” on page 306](#).

NOTE: You should save log files from the SiebSrvr\Log directory into a Zip file so that you can review them later. If you do not save them, they will be moved to the SiebSrvr\Log Archive directory and will eventually be removed when the machine is rebooted or Siebel Server services are restarted.

If you find errors that are not included in the `errors.rtf` file in the `DBSRVR_PLTFRM_ROOT` directory, contact Siebel Technical Support.

NOTE: Before you begin the upgrade of your production environment, verify that there are no repositories named “Siebel Repository” or “New Customer Repository” in your database.

- 5 Upgrade your production environment. Use the configuration utility to upgrade the Siebel database schema on your production environment, as instructed in [“Upgrading the Siebel Database Schema” on page 305](#).

- a Specify parameters you listed in the Upgrade Planning Worksheet.

- b Make sure that the following parameter is set as shown below:

```
Environment Type ="Production"
```

NOTE: You may need to change the configuration of the ODBC data source when you upgrade the Siebel database schema and the custom database schema to point to your production database before executing these scripts.

- 6 Use the configuration utility to upgrade the custom database schema on your production environment, as instructed in [“Upgrading the Custom Database Schema” on page 309](#).
 - Specify parameters you listed in the Upgrade Planning Worksheet.
- 7 Verify that no unacceptable errors are reported in the error logs. See [“Reviewing the Upgrade Log Files” on page 306](#).

Installing and Configuring the Siebel Database Server Software

Configuring the Siebel Database Server for upgrade consists of the following two sets of tasks:

- [“Installing the Siebel Database Server Software” on page 296](#)
- [“Configuring the Siebel Database Server for Upgrade” on page 297](#)

You will use the Siebel Software Configuration Utility to perform database configuration tasks.

Installing the Siebel Database Server Software

You will need to install the Release 7.5 Siebel Database Server software onto one production Siebel Server. The Siebel Database Server must be installed on a Siebel Server that has already been upgraded to Release 7.5.

NOTE: With the Siebel Enterprise Server (SES) installer, when the Siebel Server is chosen, make sure that you choose the Siebel Database Server as well.

CAUTION: In order to configure and execute Siebel Database Server procedures and maintenance scripts, you must have sufficient access to Siebel Server 7.5 directories:

You must have READ-WRITE access to `BIN` directories under Siebel Server executables in the `SIEBSRVR_ROOT` directory.

You must have READ-WRITE access to the log directories and upgrade directory.

To install the Release 7.5 Siebel Database Server software in your production environment, follow the instructions for installing database server components in *Siebel Server Installation Guide for Microsoft Windows, MidMarket Edition* in the chapter on installing the Siebel Database Server software on your RDBMS. Refer to your completed Upgrade Planning Worksheet for server names and other installation information.

NOTE: Repeat the process outlined in *Siebel Server Installation Guide for Microsoft Windows, MidMarket Edition* for your platform for installing the stored procedures and user-defined functions.

In order to have Siebel Administrator permissions, you must log in with a valid RDBMS user name and password, and this user name must have “Siebel Administrator” responsibility for the default organization.

The default administrator user name is:

SADMIN

If the user SADMIN does not exist or does not have “Siebel Administrator” responsibility for the default organization, you must contact your database administrator to establish this before you proceed with the upgrade.

This attribute can be changed to the login of another “employee” if necessary to meet your business needs.

Configuring the Siebel Database Server for Upgrade

The Siebel Software Configuration Utility generates your upgrade configuration file (UCF file):

```
master_<upgrade_option>_<upgrade_type>_<version>.ucf
```

for example,

```
master_upgrep_prod_601.ucf
```

This file will be used to run the upgrade. After your configuration is complete, this file will be created in the *SIEBSRVR_ROOT\bin* directory.

The Siebel Software Configuration Utility validates certain parameters, and will not proceed if you enter an invalid parameter. The configuration utility validates the following information:

- That you have renamed the Siebel Repository to Prior Customer Repository
- That tablespaces exist in your database
- That username and password are valid
- That tableowner and tableowner password are valid
- That the language pack is installed
- That the directories chosen exist

To change the language in which the configuration utility runs

When you launch the Siebel Software Configuration Utility, using one of the methods previously mentioned, it launches automatically in the language in which you originally chose to run the SES Installer.

You can change the language in which the configuration utility runs, if desired, from the language chosen during installation. To change the Siebel Software Configuration Utility language, refer to *Siebel Server Installation Guide for Microsoft Windows, MidMarket Edition*.

To configure the Siebel Database Server on the Windows Platform

- 1** Verify that no server tasks are running in the background.

If necessary, stop Siebel Servers and Siebel Gateway Name Server service by navigating to Start > Settings > Control Panel > Services.

- 2** Launch the Siebel Software Configuration Utility by selecting Start > Programs > Siebel Enterprise Servers 7.0 > Configure DB Server.

The Siebel Enterprise Parameters: Gateway Server Address screen appears.

- 3 Type the following values as you recorded them in your copy of [Appendix A, “Upgrade Planning Worksheet.”](#)

- **Gateway Server Address.** The alias of your Siebel Gateway Name Server (typically the machine name).
- **Enterprise Server Name.** The name of your Enterprise Server, for example, siebel.

To continue, click Next.

The Installation and Configuration Parameters: Siebel Server Directory screen appears.

- 4 Accept the default value displayed in the Siebel Server Directory field after verifying that this is where your Siebel Server is installed.

NOTE: This is the *SIEBSVR_ROOT* directory, for example,
D:\sea7xx\siebsvr.

The Installation and Configuration Parameters: Siebel Database Server Directory screen appears.

- 5 In the Siebel Database Server Directory field, verify that the default directory path displayed is the correct database server installation directory for your configuration. If it is not, use the Browse button to navigate to the correct database directory.

NOTE: This is the *DBSRVR_ROOT* directory, for example, D:\sea7xx\dsrvr.

To continue, click Next.

- 6 In the Siebel Database Server Options : RDBMS Platform screen, choose the database platform that you are upgrading.

- **IBM DB2 for Unix and Windows.**
- **Microsoft SQL Server.**
- **Oracle.**

To continue, click Next.

The Siebel Database Server Options: Siebel Database Operation menu appears.

- 7** Choose `Upgrade Database` and click `Next`.

The Upgrade the Siebel Database: Upgrade Options screen appears.

- 8** Choose `Upgrade Siebel Database Schema (upgrep)` and click `Next`.

The Upgrade Configuration Information: Environment Type screen appears.

- 9** Choose `Production` to upgrade your production environment.

To continue, click `Next`.

- 10** Choose the base version of the Siebel application from which you are upgrading.

To continue, click `Next`.

The Installation and Configuration Parameters: Language Selection screen appears.

NOTE: The language selection screen will not appear if you have only one language installed, because that language is automatically defined as the primary language. Therefore, you will skip the next step and continue with [Step 12](#).

- 11** Choose the language which was the primary language of your prior environment.

To continue, click `Next`.

NOTE: If you cannot continue, then you selected a language for which you do not have a language pack. You need to reselect the primary language used in your prior environment, and then click `Next` to continue.

The next Installation and Configuration Parameters: ODBC Data Source Name screen appears.

- 12** Accept the name of the ODBC data source displayed for verification purposes (for example, `SiebSrvr_siebel`), or enter the data source name for your installation.

NOTE: The data source is created automatically by the Siebel Server installation, using the format `SiebSrvr_EnterpriseName`. To find the name of your ODBC data source, navigate to Start > Settings > Control Panel > Administrative Tools > Data Source (ODBC). Click the System DSN tab and you will find the name of your ODBC data source.

To continue, click Next.

The Installation and Configuration Parameters: Database User Name screen appears.

- 13** Type the source user name and password for your database:

- a Database User Name.** User name of the Siebel administrator, for example, `sadmin`.
- b Database Password.** Password for the Siebel administrator.
- c Database Password (confirm).** Retype the password for confirmation.

To continue, click Next.

NOTE: The following Installation and Configuration Parameters screens are platform-specific. Which screens appear next will depend on the database platform that you are upgrading.

- 14** In the Installation and Configuration Parameters screens that appear next, complete each field with the values that you recorded on your Upgrade Planning Worksheet. The default values for each RDBMS are listed in the table below.

NOTE: Use underscores rather than spaces; these values are case-sensitive.

Field	Value
Database Table Owner	This is the account that will own the Siebel objects, for example, <code>siebel</code> . <ul style="list-style-type: none">■ DB2 UDB: Tableowner■ Microsoft SQL Server: Database Owner Login (this is the login for the owner of the database, not necessarily the default owner of the database in DBO).■ Oracle: Tableowner
Database Table Owner Password	<ul style="list-style-type: none">■ DB2 UDB: Tableowner password■ Microsoft SQL Server: the password for Database Owner Login (this is the login for the owner of the database, not necessarily the default owner of the database in DBO).■ Oracle: Tableowner password
Index Space	<ul style="list-style-type: none">■ DB2 UDB: The name you gave to your 4-KB index space for tables■ Oracle: The name you gave to your index area
4KB Table Space	<ul style="list-style-type: none">■ DB2 UDB: The name you gave to your 4-KB table space <p>NOTE: Be sure to use underscores rather than spaces</p>
8KB Table Space (DB2 UDB upgrades from 5.x only)	<ul style="list-style-type: none">■ DB2 UDB: The name you gave to your 8-KB table space
16KB Table Space (DB2 UDB only)	<ul style="list-style-type: none">■ DB2 UDB: 16 KB Table space name
32KB Table Space (DB2 UDB only)	<ul style="list-style-type: none">■ DB2 UDB: 32 KB Table space name
Table Space Name	<ul style="list-style-type: none">■ Oracle: The name you gave to your data area

After you type the value for each screen, to continue, click Next.

- 15** For all platforms, in the Upgrade Configuration Information: Database Server OS screen, choose the platform on which your database server runs.

To continue, click Next.

The Configuration Parameter Review screen appears.

- 16** Review the configuration values you entered on the previous Configuration Utility screens against the values that you recorded in your copy of [Appendix A, “Upgrade Planning Worksheet.”](#)

NOTE: Passwords are encrypted and will not appear in plain text either in the user interface or in the upgrade configuration files (UCF files). After a password is entered, it will always appear in encrypted form. If you need to use another password, you must rerun the configuration utility.

- If you need to go back to make changes, click Previous to back out until you reach the screen with the parameter you need to change. Enter the valid parameter, and then click Next until you reach the Configuration Parameter Review screen again.
 - To accept the values you input with no changes, click Finish.
- 17** A message box appears, prompting you to decide if you want to apply the configuration now or later.
- To apply the configuration now, click OK.

The Siebel Upgrade Wizard appears. To begin the upgrade of your repository, click OK. A check mark will appear beside each item as it is completed. When the status bar registers that the upgrade process is complete, click OK to exit the Siebel Upgrade Wizard.

At this stage, you have finished upgrading your repository and are ready to review the log files for errors.

- To apply the configuration later, click Cancel.

When you are ready to apply your configuration, follow the procedure [“To apply your Siebel Software configuration later” on page 304](#).

CAUTION: If you cancel or abort the upgrade process at any point, ask your database administrator to terminate the upgrade process connection to the database.

During the upgrade, the only active connections to the database should be those required for the upgrade. At no time should there be any online users connected using the Siebel eBusiness Applications.

If errors are encountered during the upgrade process, it will stop at that point. If your upgrade stops due to an error, you must carefully review the log files to make sure that your upgrade has completed successfully up to that point, and fix the error that stopped the upgrade. See [“Reviewing the Upgrade Log Files” on page 306](#). Once you have corrected the error, you may restart the upgrade, and it will continue from the last step that completed successfully. For details on how to restart, see [“Launching the Siebel Upgrade Wizard” on page 308](#).

To apply your Siebel Software configuration later

- 1 If you want to apply the configuration later, you may do this from the Windows DOS command prompt after you review the upgrade configuration file (UCF file) under the `SIEBEL_ROOT\siebsrvr\bin` directory:

```
master_<upgrade_option>_<upgrade_type>_<version>.ucf
```

for example,

```
master_upgrep_prod_601.ucf
```

- 2 After you have reviewed the UCF file, to apply the configuration, enter the following command at the DOS command prompt:

```
SIEBEL_ROOT\siebsrvr\bin\siebug.exe /m master_<upgrade_option>_<upgrade_type>_<version>.ucf
```

for example,

```
SIEBEL_ROOT\siebsrvr\bin\siebug.exe /m  
master_upgrep_prod_601.ucf
```


Upgrading the Siebel Database Schema

When you completed the previous procedure, [“Configuring the Siebel Database Server for Upgrade” on page 297](#), the Siebel Software Configuration Utility automatically launched the Siebel Upgrade Wizard.

During the production environment upgrade of Siebel Database Schema, the Siebel Upgrade Wizard performs the following major tasks:

- Upgrades the Siebel database schema to the new version
- Migrates application data to the new data model structure
- Upgrades Siebel seed data
- Loads one new repository: New Customer Repository (Release 7.5)

The Siebel Upgrade Wizard is restartable at most stages within the upgrade process. For information about restartability, see [“Restarting the Upgrade” on page 307](#).

CAUTION: If problems with your environment prevent the upgrade from restarting, you need to restore the database from the prior base version (the version which you are upgrading from). For example, environment problems may occur when table creation fails due to a database problem (insufficient storage or network problems), which cause subsequent upgrade steps to fail.

Reviewing the Upgrade Log Files

The upgrade wizard creates several log files, such as `UpgWiz.log`, `UpgWiz_01.log` (the name of the log file will increment for subsequent log files) within the `SIEBEL_ROOT\siebsrvr\LOG` subdirectory. These log files may include errors that are expected and benign: acceptable errors are documented in the `errors.rtf` file located in the `siebsrvr_root\log` directory. You must review the log files carefully for unacceptable errors.

To review the log files, see [“Reviewing the Upgrade Log Files” on page 119](#).

CAUTION: Do not proceed with the upgrade until unacceptable errors have been corrected. If you cannot correct the error, contact Siebel Technical Support or Professional Services to report the error in detail.

Troubleshooting the Siebel Database Schema Upgrade

Typical problems that may occur at this stage could result from a lack of storage space or insufficient user privileges.

Recovering from a Failed Siebel Database Schema Upgrade

If the repository upgrade fails due to insufficient space allocated on the database, you must complete the following procedures.

To recover from a failed Siebel database schema upgrade

- 1** Back up your complete set of log files, from the beginning of the upgrade process to the point at which it stopped, to another directory.
- 2** Read the `upgwiz.log` file and associated log file to determine the failure. See [“Reviewing the Upgrade Log Files” on page 306](#).
- 3** Take the necessary corrective action. Depending on the errors that you find, you may need to ask your DBA to extend the database.

- 4** Resume the upgrade wizard. It will continue from the point at which it failed.

NOTE: To resume the upgrade wizard, type the following command at the Windows DOS command prompt from the `siebsrvr_root/BIN` directory:

```
siebug.exe /m master_<upgrade_option>_<upgrade_type>_<version>.ucf
```

For example, to restart a production upgrade from Release 6.0.1 to Release 7.5, type:

```
siebug /m master_upgrep_prod_601.ucf.
```

Restarting the Upgrade

The Siebel Upgrade Wizard is restartable at most stages within the upgrade process. If the Siebel Upgrade Wizard encounters an error during the upgrade process, it will stop at that point. If your upgrade stops due to an error, you must carefully review the log files to make sure that your upgrade has completed successfully up to that point. Once you have verified this and resolved the failure, you may restart the upgrade. The upgrade will continue from the last step that completed successfully.

CAUTION: Before you restart the upgrade (after any break in the upgrade process) back up your complete set of log files, from the beginning of the process to the point at which it stopped, to another directory. This will maintain a complete record of your log files, and prevent your previous log files from being overwritten, which could prevent accurate diagnosis of the reason for the break in the upgrade.

Launching the Siebel Upgrade Wizard

To manually launch the Siebel Upgrade Wizard from the command prompt, or to restart the upgrade

- 1 Type the following command at the Windows DOS command prompt from the `SIEBEL_ROOT\siebsrvr_root\BIN` directory:

```
siebug.exe /m master_<upgrade_option>_<upgrade_type>_<version>.ucf
```

For example, to restart a production upgrade from Release 6.0.1 to Release 7.5, type the following command:

```
siebug.exe /m master_upgrep_prod_601.ucf.
```

The Siebel Upgrade Wizard screen appears, displaying the items to be completed during this phase.

- 2 To begin the upgrade of your repository, click OK.
A check mark will appear beside each item as it is completed.
- 3 When the status bar registers that the upgrade process is complete, click OK to exit the Siebel Upgrade Wizard.

You have finished upgrading your repository and are ready to review the log files for errors.

Upgrading the Custom Database Schema

The configuration utility that you ran in [“Upgrading Without a Development Environment” on page 294](#) updated the Siebel Database Schema (upgrep). You will now use the Siebel Software Configuration Utility to apply these changes to the custom database schema (upgphys) to upgrade it to the new version.

To upgrade the custom database schema (upgphys)

- 1 Verify that no server tasks are running in the background.

If necessary, stop Siebel Servers and Siebel Gateway Name Server service by navigating to Start > Settings > Control Panel > Services.

- 2 Launch the Siebel Software Configuration Utility by selecting Start > Programs > Siebel Enterprise Servers 7.0 > Configure DB Server.

The Siebel Enterprise Parameters: Gateway Server Address screen appears.

- 3 Type the following values as you recorded them in your copy of [Appendix A, “Upgrade Planning Worksheet.”](#)

- **Gateway Server Address.** The alias of your Siebel Gateway Name Server (typically the machine name).
- **Enterprise Server Name.** The name of your Enterprise Server, for example, siebel.

To continue, click Next.

The Installation and Configuration Parameters: Siebel Server Directory screen appears.

- 4 Accept the default value displayed in the Siebel Server Directory field, or type an alternate directory path for your configuration.

NOTE: This is the *SIEBSRVR_ROOT* directory, for example,
D:\sea7xx\siebsrvr.

The Installation and Configuration Parameters: Siebel Database Server Directory screen appears.

- 5 In the Siebel Database Server Directory field, verify that the default directory path displayed is the correct database server installation directory for your configuration. If it is not, use the Browse button to navigate to a different database directory.

NOTE: This is the *DBSRVR_ROOT* directory, for example, *D:\sea7xx\dbsrvr*.

To continue, click Next.

- 6 In the Siebel Database Server Options : RDBMS Platform screen, choose the database platform that you are upgrading.

- **IBM DB2 for Unix and Windows.**
- **Microsoft SQL Server.**
- **Oracle.**

To continue, click Next.

The Siebel Database Server Options: Siebel Database Operation menu appears.

- 7 Choose *Upgrade Database* and click Next.

The Upgrade the Siebel Database: Upgrade Options screen appears.

- 8 Choose *Upgrade Custom Database Schema (upgphys)* and click Next.

The Upgrade Configuration Information: Environment Type screen appears.

- 9 Choose *Production* to upgrade your production environment.

To continue, click Next.

- 10** Choose the base version of Siebel eBusiness Applications from which you are upgrading.

To continue, click Next.

The Installation and Configuration Parameters: Language Selection screen appears.

NOTE: The language selection screen will not appear if you have only one language installed, because that language is automatically defined as the primary language. Therefore, you will skip the next step and continue with [Step 12](#).

- 11** Choose the language that was the primary language of your prior environment.

To continue, click Next.

NOTE: If you cannot continue, then you selected a language for which you do not have a language pack. You need to reselect the primary language used in your prior environment, and then click Next to continue.

The next Installation and Configuration Parameters: ODBC Data Source Name screen appears.

- 12** Accept the name of the ODBC data source displayed for verification purposes (for example, `SiebSrvr_siebel`), or enter a different data source name.

NOTE: The data source is created automatically by the Siebel Server installation, using the format `SiebSrvr_EnterpriseName`.

To find the name of your ODBC data source, navigate to Start > Settings > Control Panel > Administrative Tools > Data Source (ODBC). Click the System DSN tab and you will find the name of your ODBC data source.

To continue, click Next.

The Installation and Configuration Parameters: Database User Name screen appears.

13 Type the source user name and password for your database:

- **Database User Name.** User name of the Siebel administrator, for example, `sadmin`.
- **Database Password.** Password for the Siebel administrator, for example, `sadmin`.
- **Database Password (confirm).** Retype the password for confirmation.

To continue, click Next.

NOTE: The following Installation and Configuration Parameters screens are platform-specific. Which screens appear next will depend on the database platform that you are upgrading.

- 14** In the Installation and Configuration Parameters screens that appear next, complete each field with the values that you recorded on your Upgrade Planning Worksheet. The default values for each RDBMS are listed in the table below.

NOTE: Use underscores rather than spaces; these values are case-sensitive.

Field	Value
Database Table Owner	<p>This is the account that will own the Siebel objects, for example, <code>siebel</code>.</p> <ul style="list-style-type: none"> ■ DB2 UDB: Tableowner. ■ Microsoft SQL Server: Database Owner Login (this is the login for the owner of the database, not necessarily the default owner of the database in DBO). ■ Oracle: Tableowner.
Database Table Owner Password	<ul style="list-style-type: none"> ■ DB2 UDB: Tableowner password. ■ Microsoft SQL Server: the password for Database Owner Login (this is the login for the owner of the database, not necessarily the default owner of the database in DBO). ■ Oracle: Tableowner password.
Index Space	<ul style="list-style-type: none"> ■ DB2 UDB: The name you gave to your 4-KB index space for tables. ■ Oracle: The name you gave to your index area.
4KB Table Space	<ul style="list-style-type: none"> ■ DB2 UDB: The name you gave to your 4-KB table space.
8KB Table Space (DB2 UDB upgrades from 5.x only)	<ul style="list-style-type: none"> ■ DB2 UDB: The name you gave to your 8-KB table space.
16KB Table Space (DB2 UDB only)	<ul style="list-style-type: none"> ■ DB2 UDB: 16 KB Table space.
32KB Table Space (DB2 UDB only)	<ul style="list-style-type: none"> ■ DB2 UDB: 32 KB Table space.
Table Space Name	<ul style="list-style-type: none"> ■ Oracle: The name you gave to your data area.

After you type the value for each screen, to continue, click Next.

- 15** For all platforms, in the Upgrade Configuration Information: Database Server OS screen, choose the platform on which your database server runs; for example, Windows.

To continue, click Next.

The Configuration Parameter Review screen appears.

- 16** Review the configuration values you entered on the previous Configuration Utility screens against the values that you recorded in your copy of [Appendix A, “Upgrade Planning Worksheet.”](#)

NOTE: Passwords are encrypted and will not appear in plain text either in the user interface or in the upgrade configuration files (UCF files). After a password is entered, it will always appear in encrypted form. If you need to use another password, you must re-run the configuration utility.

- If you need to go back to make changes, click Previous to back out until you reach the screen with the parameter you need to change. Enter the valid parameter, and then click Next until you reach the Configuration Parameter Review screen again.
- To accept the values you input with no changes, click Finish.

- 17** A message box appears, prompting you to decide if you want to apply the configuration now or later.

- To apply the configuration now, click OK.

The Siebel Upgrade Wizard appears. To begin the upgrade of your Custom Database Schema, click OK. A check mark will appear beside each item as it is completed. When the status bar registers that the upgrade process is complete, click OK to exit the Siebel Upgrade Wizard.

At this stage, you have finished upgrading your Custom Database Schema and are ready to review the log files for errors.

- To apply the configuration later, click Cancel.

When you are ready to apply your configuration, follow the procedure “[To apply your Siebel Software configuration later](#)” on page 315, “[To apply your Siebel Software configuration later](#).”

CAUTION: Some long-running operations, when canceled, may not sever all processes on the database side. If you cancel or abort the upgrade process at any point, ask your database administrator to terminate your connection to the database by using the *kill* command.

If your upgrade does not complete successfully, work with your database administrator to verify that there are no other users currently logged on to the database. If there are, ask your database administrator to terminate all connections, then relaunch the configuration wizard.

If errors are encountered during the upgrade process, it will stop at that point. If your upgrade stops due to an error, you must carefully review the log files to make sure that your upgrade has completed successfully up to that point, and fix the error that stopped the upgrade. See [“Troubleshooting the Custom Database Schema Upgrade” on page 316](#). Once you have corrected the error, you may restart the upgrade, and it will continue from the last step that completed successfully. For details on how to restart, see [“Launching the Siebel Upgrade Wizard” on page 308](#).

To apply your Siebel Software configuration later

- 1 If you want to apply the configuration later, you must first review the upgrade configuration file (UCF file) under the *SIEBEL_ROOT\siebsrvr\bin* directory:

```
master_<upgrade_option>_<upgrade_type>_<version>.ucf
```

for example,

```
master_upgphys_prod_601.ucf
```

- 2 After you have reviewed the UCF file, to apply the configuration, enter the following command at the DOS command prompt:

```
SIEBEL_ROOT\siebsrvr\bin\siebupg.exe /m master_<upgrade_option>_<upgrade_type>_<version>.ucf
```

for example,

```
SIEBEL_ROOT\siebsrvr\bin\siebupg.exe /m  
master_upgphys_prod_601.ucf
```

Troubleshooting the Custom Database Schema Upgrade

Typical problems that might occur during the upgrade of your custom database schema (upgphys) could result from a lack of storage space or insufficient user privileges.

Reviewing the Upgrade Log Files

The upgrade wizard creates several log files, such as `UpgWiz.log`, `UpgWiz_01.log` (the name of the log file will increment for subsequent log files) within the `SIEBEL_ROOT\siebsrvr\LOG` subdirectory. These log files may include errors that are expected and benign: acceptable errors are documented in the `errors.rtf` file located in the `siebsrvr_root\log` directory. You must review the log files carefully for unacceptable errors.

To review the log files, see [“Reviewing the Upgrade Log Files” on page 119](#).

CAUTION: Do not proceed with the upgrade until unacceptable errors have been corrected. If you cannot correct the error, contact Siebel Technical Support or Professional Services to report the error in detail.

Restarting the Upgrade

The Siebel Upgrade Wizard is restartable at most stages within the upgrade process. If the Siebel Upgrade Wizard encounters an error during the upgrade process, it will stop at that point. If your upgrade stops due to an error, you must carefully review the log files to make sure that your upgrade has completed successfully up to that point. Once you have verified this and resolved the failure, you may restart the upgrade. The upgrade will continue from the last step that completed successfully.

CAUTION: Before you restart the upgrade (after any break in the upgrade process) back up your complete set of log files, from the beginning of the process to the point at which it stopped, to another directory. This will maintain a complete record of your log files, and prevent your previous log files from being overwritten, which might prevent accurate diagnosis of the reason for the break in the upgrade.

To manually launch the Siebel Upgrade Wizard from the command prompt, or to restart the upgrade

- 1 Type the following command at the Windows DOS command prompt from the `SIEBEL_ROOT\siebsrvr_root\BIN` directory:

```
siebug.exe /m master_<upgrade_option>_<upgrade_type>_<version>.ucf
```

For example, to restart a production upgrade from Release 6.0.1 to Release 7.5, type the following command:

```
siebug.exe /m master_upgphys_dev_601.ucf.
```

The Siebel Upgrade Wizard screen appears, displaying the items to be completed during this phase.

- 2 To begin the upgrade of your repository, click OK.
A check mark will appear beside each item as it is completed.
- 3 When the status bar registers that the upgrade process is complete, click OK to exit the Siebel Upgrade Wizard.

You have finished upgrading your repository and are ready to review the log files for errors.

CAUTION: Do not proceed further with the upgrade until such errors have been analyzed and the necessary corrective action taken.

Adding New License Keys

With the new version of Siebel eBusiness Applications, you received one or more license keys that must be added to the production database. You must add all of the new license keys to enable the new versions of the Siebel products, which you will use in the next steps of the upgrade process.

To add new license keys

- 1** Start Siebel Tools version 7.5 from a production workstation and log on to the database server as the Siebel administrator.
- 2** If you are not prompted for a new license key before Siebel Tools starts, navigate to Help > Technical Support > License Keys.
- 3** Add your new license keys.

Production Environment Postupgrade Tasks

9

This chapter describes the tasks which you need to perform after the upgrade of your Siebel eBusiness Applications production environment. [Table 37 on page 320](#) illustrates the sequence of postupgrade steps.

After you have successfully completed the upgrade steps in [Chapter 8, “Upgrading the Production Environment”](#) continue with the post–upgrade tasks listed in [Table 37 on page 320](#).

Table 37. Siebel Production Environment Post Upgrade Tasks

Post Upgrade Tasks	
1	Drop DB2 8-KB table spaces and buffer pools. See “Dropping DB2 8-KB Table Spaces and Buffers” on page 321.
2	Manually archive the log files. See “Manually Archiving the Log Files” on page 321.
3	Generate reporting relationships. See “Generating Reporting Relationships” on page 323.
4	Update file system directory. See “Updating the File System Directory” on page 322.
5	Update file system attachments. See “Updating File System Attachments” on page 322.
6	Validate the upgrade. See “Validating the Upgrade” on page 326.
7	Run statistics. See “Run Statistics” on page 326.
8	Reorganize the P1 index for DB2 UDB. See “Reorganize the P1 Index for DB2 UDB” on page 326.
9	Reset the database server configuration parameters. See “Reset Database Server Configuration Parameters” on page 326.
10	Review postupgrade tasks and considerations for specific applications. See “Additional Postupgrade Tasks for Specific Applications” on page 327.
11	Upgrade your encryption method to RC2 before you deploy your upgraded Siebel eBusiness Application. See “Upgrading to RC2 Encryption” on page 330.
12	Set up your environment to support global time zone. See “Setting Up Your Environment to Support Global Time Zone” on page 331.
13	Review planning considerations for migration from a non-Unicode code page to Unicode. See “Migrating to Unicode” on page 332.
14	Upgrade Mobile and Connected Clients. See “Upgrading Mobile and Connected Clients” on page 332.

Postupgrade Tasks for All RDBMS Production Environments

This section describes postupgrade tasks for production environments.

Dropping DB2 8-KB Table Spaces and Buffers

If you are upgrading a database under Windows or UNIX, you should now drop your 8-KB table space, 8-KB temporary table space, and 8-KB buffer pool. Before dropping your 8-KB table space, check for the existence of any tables in it by running the following SQL statement:

```
select name from sysibm.systables where TBSPACE='TBS_8K'
```

CAUTION: Check your 8-KB table space to make sure that all objects have been migrated to your new 16-KB table space before dropping the 8-KB table space. Otherwise, you will lose this data.

Manually Archiving the Log Files

After a successful installation and upgrade, you must save and archive the log files and remove the `state.log` file from the current install directory, in the subdirectory for the driver (`SIEBEL_ROOT/upgrade/PROCESS`). This is a manual process.

CAUTION: If you do not archive and remove the `state.log` file, when you run subsequent runs of the same process against the same environment, the configuration utility will produce erroneous results.

By default, only nine (9) log files are retained for subsequent retries of the upgrade wizard. After nine log files have been created, when the upgrade wizard is rerun, it will overwrite log files beginning with the earliest one created and recycle the rest as necessary.

The number of log files retained can be increased by resetting the `siebel_log_archive` Windows environment variable to 20, for example, to retain twenty (20) log files.

Updating the File System Directory

In your previous installation, the file system had a flat structure with one directory. The Release 7.x file system contains subdirectories. During the upgrade of your production environment, Siebel Anywhere will look for certain files in the file system subdirectories, but these files only exist in the root file system directory.

Therefore, you need to create the appropriate subdirectories in your previous installation directory structure and then copy the required files from the root file system directory to the new subdirectories.

To make file attachments accessible, see [“Updating the File System Directory” on page 174](#).

Updating File System Attachments

During the Siebel database upgrade, data from the inactive table `S_EMPLOYEE` was migrated to `S_CONTACT`, `S_USER`, `S_EMP_PER` and data from `S_ORG_INT` was migrated to `S_ORG_EXT`. Attachments in `S_ORG_INT_ATT` were migrated to `S_ACCNT_ATT` and attachments in `S_EMPLOYEE_ATT` were migrated to `S_CONTACT_ATT`.

NOTE: If your upgrade to Release 7.5 was from Release 7.0.3 or 7.0.4, files from `S_WEB_CNTNT` were migrated to `S_CB_ASSET_VER`.

Data migration affects the physical file structure of the Siebel File System. Therefore, you need to run a utility to rename file attachments which correspond to inactive tables for Release 7.0 so that they will be accessible by Siebel eBusiness Applications. For example, this utility will copy and rename all files named `S_EMPLOYEE*.SAF` to `S_CONTACT*.SAF` and all files named `S_ORG_INT_ATT*.SAF` to `S_ACCNT*.SAF` so that they correspond to new table names.

To update file system attachments, see [“Updating File System Attachments” on page 175](#).

Generating Reporting Relationships

In Release 7.x, there are three visibility hierarchies—position, organization, and access groups. These hierarchies are denormalized and maintained in the table S_PARTY_RPT_REL. These denormalized hierarchies are necessary for executing visibility modes that go up or down a hierarchy. For example:

- **Manager view mode.** “My Team’s Accounts View” displays all accounts on which a manager and their subordinates are working.
- **Sub-Organizations view mode.** “All Contacts across My Organizations View” displays all contacts that are associated to either my organization or any of my organization’s sub-organizations.

The generate reporting relationships process rebuilds the denormalized relationships in the S_PARTY_RPT_REL table so that the hierarchical view modes will display the correct information. The basic operation of the function is to empty the S_PARTY_RPT_REL table and then walk through each S_PARTY record to recreate the denormalized hierarchical structures in the table. This process generates a large number of transactions for Siebel Remote users.

NOTE: This operation is time and cpu/memory intensive. The process may take several minutes, depending on the size and complexity of your organizational structures. You should not perform this when you are running other memory-intensive processes.

The Generate Reporting Relationships process needs to be executed after the upgrade to Release 7.x and whenever the denormalized hierarchy structure (S_PARTY_RPT_REL) becomes out of sync with the data in the normalized tables (S_PARTY). The following situations may cause these tables to become out of sync:

- After upgrading to Release 7.x, the organizational hierarchy (even if there is only one organization) must be established to maintain appropriate visibility in the views cited above.
- When EIM is used to import or update any of the hierarchies (positions, organizations, or access groups).

The standard Release 7.x configuration includes the Generate Reporting Relationships feature as a hidden button on the Position List Applet NB. You need to go through Siebel Tools configuration to expose this button.

To expose the Generate Reporting Relationships button

- 1** Open Siebel Tools and navigate to the applets folder.
- 2** Find and select the Position List Applet NB applet record.
- 3** Right click on the record and select Edit Web Layout.
- 4** Drag the “GenReportRel” button from the Controls/Columns window into one of the button placeholders in the applet layout (that is, one of the empty “x” placeholders in the blue header area of the applet layout).
- 5** Repeat this step for the three different modes (Base, Edit, and Edit List) in which the applet can be displayed. The easiest way to switch between the different modes is to just use the Mode drop-down that appears in the Web Controls toolbar of Siebel Tools. After you have drag-and-dropped the button into all three modes of the applet layout, close the layout editor and save your changes.
- 6** Recompile the applet into your existing `siebel.srf` as used by the Web client.
- 7** Launch the Web client or Dedicated Web client using the SRF compiled in the previous step so that the Generate Reporting Relationships button can be invoked.

NOTE: Siebel Systems recommends that you do not make this button available in the standard `siebel.srf` file used by your organization in order to preserve control over who can press this button and when it can be pressed.

To generate reporting relationships

- 1** If you have an active Siebel Remote environment, confer with a Siebel Systems Administrator. The Administrator should arrange for the Transaction Processor to be paused before performing this procedure.
- 2** Choose Group Administration under Site Map and navigate to the Positions view in the Siebel Web client application. Click the Generate Reporting Relationships button in the Position List Applet NB. Please note that generating the reporting relationship may cause a large number of Siebel Remote transactions to be generated.
- 3** When this has completed, restart the Transaction Processor.

Setting Visibility Modes for Access Control

NOTE: You only need to perform this procedure if you are upgrading *without* a development environment. If you have a development environment, you already performed this procedure as a postupgrade task for the development environment.

Before you deploy the upgraded configuration in your test or production environment, you need to make a decision about what type of visibility to use on certain business components, views, and applet picklists. This requires some evaluation. Certain areas of the Release 7.x default configuration use Catalog visibility.

To set visibility modes for access control, see [“Setting Visibility Modes for Access Control” on page 183](#).

Producing a New Custom Configuration File

NOTE: You only need to perform this procedure if you are upgrading *without* a development environment. If you have a development environment, you already performed this procedure as a postupgrade task for the development environment.

You must now compile a new Siebel repository file (.srf). For more information, see [“To compile a new .srf file” on page 188](#).

Validating the Upgrade

Run the following utilities to verify that your upgrade was successful.

- **DBCHCK.** This utility verifies that the physical schema is in sync with the logical schema (dictionary). If tables are missing or mismatched in either the logical (dictionary) or physical schema, they are reported in the dbchk.log file generated by DBCHCK.
- **DICTUTL.** This utility verifies that all dock object and rule definitions are correct.

For information about how to run these utilities, see [“Validating the Upgrade” on page 190](#).

Run Statistics

If your RDBMS is DB2 UDB or Microsoft SQL, you should run full statistics on all tables used in your environment.

- If your RDBMS is DB2 UDB, run `updatestats.sql`, found under `DBSERVER_ROOT/DB2UDB`.

Reorganize the P1 Index for DB2 UDB

After your upgrade to Release 7.0, you need to verify that the table S_EVT_ACT is reorganized on the P1 index. This happens because, in the repository, the append mode flag is set. If S_EVT_ACT is not reorganized by the P1 index, then Siebel Remote performance will be negatively affected.

Reset Database Server Configuration Parameters

After you complete your upgrade, you need to reset your database server configuration to installation settings. See *Siebel Server Installation Guide for Microsoft Windows, MidMarket Edition*.

Additional Postupgrade Tasks for Specific Applications

This section describes additional postupgrade tasks and considerations for the following applications:

- Call Center
- Siebel Purchase Orders
- Quotes
- Workflow

Call Center

NOTE: If you are upgrading from Release 7.0.x to Release 7.5, you do not need to perform this procedure.

During the upgrade to Release 7.x, employee and contact data were moved to the same tables. As a result of this change, your database will contain duplicate logins for Contact user login names.

In order to prevent the creation of duplicate logins which may occur due to this change, the Release 7.x upgrade will append the `ROW_ID` to duplicate logins. You need to resolve your user logins after the upgrade, or users may be unable to log in. For example, contact user login names will appear concatenated with their row ID.

To locate user logins that require resolution of duplicates

- 1** Open your Call Center application and navigate to Site Map > User Administration.
- 2** For each User Administration view (Employees, Persons, and Users) query the User ID field for login name = * + *. This query will bring up all names that are appended with “ + <ROW_ID> ”.
- 3** Repeat this procedure for each User Administration view (Employees, Persons, and Users).

Siebel Purchase Orders

After you have upgraded to Release 7.5, you need to run a utility to update the Transaction Amount field in Payment Lines that have Purchase Order as the payment method. The utility requires the .srf that you compiled after upgrading your development environment.

The utility performs the following steps:

- 1** Creates a new Order Entry business object.
- 2** Creates an Order Entry business component and a Payments business component.
- 3** Checks all order records and looks at the Payment Method of corresponding Payment Lines.
- 4** If the Payment Method of a Payment Line is Purchase Order, it updates the Transaction Amount field to Order Total.

The utility requires the following parameters:

- `username`. Siebel user login name
- `password`. Siebel login password
- `lang`. Language used
- `cfg file`. The configuration file used to launch the application
- `data source`. Date source used from the .cfg file

To run the utility from the command line, enter

```
Pmntupgd /u <username> /p <password> /l <lang> /c <cfg file> /d <data source>
```


Quotes

After upgrade, you need to run the following statement to fix an issue where quote items had trailing spaces added accidentally. This script is mandatory.

```
update S_QUOTE_ITEM
setROW_ID = rtrim(ROW_ID)
,   ROOT_QUOTE_ITEM_ID = rtrim(ROOT_QUOTE_ITEM_ID)
,   PAR_SQ_ITEM_ID = rtrim(PAR_SQ_ITEM_ID)
,   PORT_VALID_PROD_ID = rtrim(PORT_VALID_PROD_ID)
,   PROD_PORT_ID = rtrim(PROD_PORT_ID)
,   INTEGRATION_ID = rtrim(INTEGRATION_ID)
;

commit
;
```

Siebel Workflow

Perform the following postupgrade steps after a successful upgrade of Siebel Workflow to Release 7.5.

- Change inbound workflows that contain a “String” type process property to pass the value into type `Binary`; otherwise, the workflow will give the following error message:

```
Output argument '<Value>' in step 'Read from File' contains data
that cannot be passed to string type property 'InputXML'. Data
type: 'MEMBLOCK'; String representation of data body: '<?xml
version="1.0" encoding="UTF-8"><?'
```

- After an upgrade from Release 6.x to 7.5, manually change the name of the “EAI MQSeries Transport” business service to the name “EAI MQ Series Server Transport”; otherwise, the workflow will give the following error message:

```
[1] Unable to create the Business Service 'EAI MQSeries Transport'
[2] Could not find 'Class' named 'EAI MQSeries Transport'. This
object is inactive or nonexistent.
```

Upgrading to RC2 Encryption

CAUTION: If you have data that was encrypted using the Release 6.x or 7.0.x standard encryptor, and you need to be able to read this encrypted data in Release 7.5, you must upgrade your encryption method after successfully upgrading but before deploying the application.

Release 7.5 delivers a new default encryption method that is based on the RC2 standard. The previous default encryption method (called the standard encryptor) is no longer supported, and data that used the standard encryptor will not be read properly by 7.5 applications unless you upgrade your encryption method to RC2. Use the Encryption Upgrade Utility to convert unencrypted data and data that was encrypted using the standard encryptor to the RC2 encryption method.

CAUTION: Using a non-RC2 encryption method in a Unicode environment will result in irrecoverable data loss.

To upgrade your encryption method, verify the prerequisites and follow the procedures under [“Upgrading Your Encryption Method” on page 199](#).

Postupgrade Tasks for Global Deployments

The following procedures are for global deployments, deployments that intend to become global in the future, and deployments that wish to take advantage of global deployment features:

- [Setting Up Your Environment to Support Global Time Zone](#)
- [Migrating to Unicode](#)

Thoroughly read *Global Deployment Guide, MidMarket Edition* before you internationalize any environment.

Setting Up Your Environment to Support Global Time Zone

Global deployments typically span multiple geographies and have users working in several different time zones. The global time zone feature converts and stores date and time data using the Universal Time Coordinated (UTC) standard, which is equivalent to Greenwich Mean Time, but without daylight savings time. This feature allows you to track dates and times in a common format across time zones.

NOTE: Although enabling your environment for global time zone is optional in Release 7.x, it is strongly recommended that you operate your production environment with global time zone enabled.

Perform one of the following procedures, as appropriate to your upgrade path:

- **For upgrades from Release 6.x to 7.5.** See [“Enabling Global Time Zone” on page 203](#) for information about enabling global time zone support after an upgrade.
- **For upgrades from Release 7.0.x to 7.5.** See [“Upgrading UTC Delta Columns” on page 203](#) for information about upgrading 7.0.x data for columns that were *not* UTC-enabled in Release 7.0.x that changed to UTC-enabled in Release 7.5.

For detailed information about UTC and global deployments for Siebel eBusiness Applications, see *Global Deployment Guide, MidMarket Edition*.

Migrating to Unicode

After you have upgraded your Siebel eBusiness Application to Release 7.5, you can migrate your upgraded database from a non-Unicode code page (or character set) to Unicode. Migration to Unicode is optional and requires the assistance of Siebel Expert Services or Siebel Professional Services.

For prerequisites and more information, see [“Migrating to Unicode” on page 205](#). For detailed information about Unicode and global deployment for Siebel eBusiness Applications, see *Global Deployment Guide, MidMarket Edition*.

CAUTION: Use of a non-RC2 encryption method in a Unicode environment might result in irrecoverable data loss. To upgrade your encryption method before you proceed with the Unicode migration, see [“Upgrading to RC2 Encryption” on page 330](#).

Upgrading Mobile and Connected Clients

After a successful upgrade of Siebel eBusiness Applications, you are ready to upgrade mobile and connected clients.

To upgrade Siebel client software on connected and mobile user workstations

- 1** Uninstall the previous version of the Siebel client application.
- 2** Install the Release 7.5 version of the Siebel client application. You can use either of the following methods for the installation:
 - Use the Release 7.5 installation CD.
 - Use a custom installer created using the Siebel Packager utility. For more information about the Packager utility, see *Siebel Web Client Administration Guide, MidMarket Edition*.

NOTE: See *Siebel Web Client Administration Guide, MidMarket Edition* for detailed instructions about how to uninstall or install Siebel clients.

Upgrade Planning Worksheet

A

This chapter presents a sample worksheet for planning an upgrade.

Worksheet

Table 38 is a sample worksheet for planning an upgrade.

Table 38. Worksheet for Upgrade Planning

Information Needed	Value at Your Site	Comment
Siebel tableowner account username and password		The database account that owns the Siebel eBusiness Applications database objects. For DB2, Microsoft SQL Server, and Oracle, SIEBEL is the default username and password.
Siebel Administrator username and password		This must be a valid RDBMS username and password, and it must be set up as a Siebel Systems employee. The employee record must have the “Siebel Administrator” responsibility. SADMIN is the default administrator username and password.
ODBC data source name		This is the name of the ODBC data source used to connect to the Siebel eBusiness Applications database server. The default is "SAS_enterprise_server_siebel_server".
Application Server root directory		This is the root directory of your existing Siebel Server software installation.
File System directory name		This is the path from the Siebel Server to the Siebel File System, which may use a share name, drive mapping, or other method, depending on your Siebel File System.
Siebel Data Segment Name		This is the name of the table space or segment on which the Siebel eBusiness Applications tables are stored in Oracle, Microsoft SQL Server, and DB2 UDB installations.
Siebel Index Segment Name		This is the name of the table space or segment on which the Siebel eBusiness Applications indexes are stored in Oracle, Microsoft SQL Server, and DB2 UDB installations.

Index

A

- applet Web templates
 - merging (development environment) 125
- Application Updates Attribute list applet, accessing and description 152
- Autogrowth parameter, enhancing upgrade performance, MS SQL Server 97

B

- backing up, development database 63, 251
- buscomps
 - migrating, about and location of list 170
- business components
 - copying, guidelines for 135

C

- CLI packages, increasing 77, 262
- client installation directory, character name limitation for path name 174, 277
- configuration files, producing custom configuration file in development environment 188
- conflict resolution
 - development environment, examples 153
 - development environment, reviewing object attribute differences 152
- Customer Repository
 - backing up 187
 - conflict resolution, changing (development environment) 153
 - license key, adding (development environment) 123, 318
 - merge process outcome, reviewing 150

- merge, preparing for 155
- merge.txt file, reviewing for error 149
- merging procedure 148
- object attribute differences, reviewing 152
- object differences, reviewing 151
- obsolete tables, reapplying custom extensions 177
- repositories in development database, table of 146
- repository merge prerequisites 147
- repository merge, improving performance (development environment) 144
- repository merge, steps to identify objects 143
- custrep.dat 294
- custrep.dat file, saving 189

D

- database administration
 - See also* language characteristics, setting
- database server configuration
 - verifying 68, 254
- DB2 database
 - configuration parameters, setting 78, 263
 - configuration parameters, setting and table of 80, 266
 - manager configuration parameters, table of 78, 264
- DB2 UDB database
 - .dll files, about creating and copying to server 83, 269
 - repository, creating 16-KB tablespaces and buffers 86, 87, 88, 272, 273, 274

- stored procedures and user-defined functions, installing 82, 268
- tablespace page size requirements, determining (procedure) 88, 274
- UNIX or Windows, creating 32-KB tablespace and buffers 87, 273
- UNIX or Windows, creating additional tablespaces and buffers 85, 271
- DB2 UDB, expected database growth (table) 69, 255
- See also *individual development environment* and *production environment entries*
- DB2 Universal Database, optimizing performance
 - database configuration parameters, setting 78, 263
 - database manager configuration parameters, table of 78, 264
 - DB2 database configuration parameters, setting and table of 80, 266
 - db2set parameters, setting and table of 79, 265
 - ODBC statement handles, increasing the number of 77, 262
 - db2set parameters, setting and table of 79, 265
- DBSRVR_PLTFRM_ROOT, about 17
- .dll files, about creating and copying to server 83, 269
- dedicated users, preparing for the upgrade (production environment) 65, 252
- development database
 - backing up 63, 251
- development environment, post-upgrade tasks
 - See also *individual development environment entries*
 - database template file, regenerating 189
 - DB2 8-KB tablespaces and buffers, dropping 173, 321
 - developers or Siebel Tools clients, extracting 189
 - EIM mappings, deleting custom mappings 187
 - EIM mappings, resolving duplicates 186
 - log files, about saving 189
 - log files, manually archiving 173, 321
 - new custom configuration file, producing 188
 - new custom repository, backing up 187
 - obsolete tables, reapplying custom extensions 177
 - S_SRC_PAYMENT table, reviewing columns added to 34
 - tasks, table of 172
 - testing upgraded configuration 189
 - upgrade, validating 190
- development environment, pre-upgrade tasks
 - backing up, about 63, 251
 - changes, checking-in pending changes 63
 - database file size, about increasing 97, 282
 - File System directory, upgrading 174, 322
 - indexes, about preserving custom indexes 92, 278
 - interface table data, about saving 64, 251
 - RDBMS server software, about upgrading 66, 253
 - tasks, sequence of (table) 59, 247
 - triggers, about customized triggers and upgrading 66, 253, 279
 - Upgrade Planning Worksheet, completing before starting upgrade 59
- development environment, upgrading
 - See also *individual development environment entries*
 - Customer Repository, preparing for merge 155
 - development servers, list of upgrading tasks 106, 289
 - EIM temporary columns, generating 155

- repository merge, performing 143
- Siebel Database Server software,
 - installing and configuring 109, 292, 293, 296, 297
- Siebel Tools, installing 105
- upgrade tasks, table of 104, 288
- Windows, upgrading the servers 107, 290
- development repository, renaming existing repository 96, 281
- .dll files, about creating and copying to server 83, 269
- dock object visibility rules
 - repository upgrade, about (development environment) 97, 282
- DOS command prompt
 - Siebel Upgrade Wizard (development environment), manually launching 122, 169, 308, 317

E

Enterprise Integration Manager (EIM). *See individual development environment and development environment entries*

F

File System

- development environment, upgrading directory 174, 322

From 88, 274

G

Global Time Zone

- about 202, 331

I

indexes

- preserving custom indexes on tables (development environment), about 92, 278

interface table data

- development environment, saving 64, 251

L

labels and fields

- merging (development environment) 125

license keys

- development environment, adding new keys 123, 318

log files

- See also* troubleshooting
- archiving manually 173, 321
- repository upgrade log files, reviewing (development environment) 119, 121
- repository upgrade, about saving 189

M

merge.txt file, reviewing for errors 149

Microsoft SQL Server database. *See* MS SQL Server database

mobile clients

- production environment upgrade, preparing for 65, 252

MS SQL Server database

- upgrading, about 71, 256

MS SQL Server, enhancing upgrade performance 97

MS SQL Server, expected database growth (table) 69, 255

mstrep.dat 294

multilingual seed data

- Siebel Database Server for DB2, installing 110

N

New Customer Repository, about rerunning repository merge after entering Override flag 153

O

object differences

- development environment, reviewing 151

- objects
 - descendents or copies, viewing 134
- ODBC data source 294, 295
- ODBC drivers
 - development environment, avoiding conflicts with 107, 290
- ODBC statement handles, increasing the number 77, 262
- Oracle database
 - upgrading, steps before upgrading 73, 259
- Oracle database, expected database growth during upgrade (table) 69, 255
- Override flag
 - conflict resolution (development environment), using to change 153

P

- physical schema, upgrading (development environment)
 - customized business components configurations, about migrating and list of location 170
 - database (physical), upgrading 161, 309
- procedure code, installing stored code 82, 268
- production environment, pre-upgrade tasks
 - mobile and dedicated users, preparing for upgrade 65, 252
 - Siebel Anywhere, about using to upgrade 245

R

- Release 5.x DB2 UDB
 - creating 16-KB tablespaces and buffers 86, 87, 88, 272, 273, 274
- Release 5.x, DB2 UDB
 - creating 16-KB tablespaces and buffers 86, 87, 88, 272, 273, 274
- repositories, upgrading
 - See also* upgrading inheritance
 - merge process, listing results 152

- repositories. *See* Siebel Repository
- repository
 - exprep.ksh 187
- repository merge, performing (development environment)
 - about and steps to identify object differences 143
 - conflict resolution, examples of 153
 - development database repositories (table) 147
 - license keys, adding 123, 318
 - merge process, reviewing overall outcome 150
 - merge.txt file, reviewing for errors 149
 - merging procedure 148
 - object attribute differences, reviewing 152
 - object differences, reviewing 151
 - troubleshooting 144
- repository upgrade log files, reviewing development environment 119, 121
- repository, upgrading (development environment)
 - list of tasks 118, 305
 - Siebel Upgrade Wizard, launching 122, 169, 308, 317
 - troubleshooting 121, 168, 306, 316

S

- S_SRC_PAYMENT table, reviewing columns added to 34
- schema. *See* physical schema, upgrading (development environment); physical schema, upgrading (production environment)
- seed data
 - Siebel Database Server for DB2, installing multilingual seed data 110
- servers, upgrading (development environment)
 - about and tasks, list of 106, 289
 - Windows platform, upgrading all servers on 107, 290

- Siebel Administrator
 - permissions, about and login name (development environment) 110, 292, 297
- Siebel Anywhere
 - production client software, about using to upgrade 245
- Siebel Application Upgrader
 - merge process, listing results 152
- Siebel Database Server for DB2, post-installation tasks
 - multilingual seed data, installing 110
- Siebel Database Server software (development environment)
 - configuring for upgrade 110, 297
 - installing 109, 292, 293, 296
 - Kill command, using to abort or cancel upgrade 117, 304
 - Siebel Server 7.0 directories, access to 109, 292, 296
 - Siebel Software Configuration Utility, about changing the language 111, 298
- Siebel Database, expected database growth (table) 69, 255
- Siebel eBusiness Applications
 - Windows, uninstall failure 105
- Siebel Enterprise Server
 - development environment, about installing Siebel Database Server Software 109, 291, 292, 296
- Siebel Gateway
 - See also* development environment, upgrading; production environment, upgrading
 - uninstalling (development environment) 108
- Siebel Remote
 - database template file, regenerating 189
- Siebel Repositories
 - DB2 UDB on UNIX or Windows, creating 32-KB tablespace and buffers 87, 273
 - DB2 UDB on UNIX or Windows, creating additional tablespaces and buffers 85, 271
 - DB2 UDB, creating 16-KB tablespaces and buffers 86, 87, 88, 272, 273, 274
- Siebel Repository
 - See also* Customer Repository
 - conflict resolution, changing (development environment) 153
 - development environment, preparing for upgrade 95, 281
 - development repository, renaming existing repository 96, 281
 - failed repository upgrade, recovering (development environment) 121, 306
 - license key, adding (development environment) 123, 318
 - merge process outcome, reviewing (development environment) 150
 - merge.txt file, reviewing for errors (development environment) 149
 - merging procedure (development environment) 148
 - new custom repository, backing up 187
 - object attribute differences, reviewing (development environment) 152
 - object differences, reviewing (development environment) 151
 - obsolete tables, reapplying custom extensions 177
 - repositories in development database, table of 146
 - repository merge prerequisites 147
 - repository upgrade log files, reviewing (development environment) 119, 121
 - restarting problems, troubleshooting 118, 305
- Siebel Upgrade Wizard, manually launching 122, 169, 308, 317

- visibility and dock object visibility rules, during upgrade (development environment) 97, 282
- Siebel Servers
 - See also* development environment, upgrading; production environment, upgrading
 - uninstalling earlier versions 107, 290
- Siebel Software Configuration Utility
 - language, about changing 111, 298
- Siebel tablespaces
 - UB2 UDB, about using database managed tablespaces (DMS) 58, 246
- Siebel Tools
 - See also* Siebel Repository installing 105
- Siebel Upgrade Wizard
 - development environment, manually launching 122, 169, 308, 317
- Siebel Workflow Manager, about running 63
- SIEBEL_CLNT_ROOT, about 17
- SQL Anywhere
 - template database, about regenerating 189
- .srf file. *See* configuration file
- state.log file, about archiving and removing 173, 321
- strings
 - migrating (development environment) 125
- T**
- tables
 - custom indexes, preserving (development environment) 92, 278
- tables, obsolete
 - reapplying custom extensions 177
- tablespaces
 - DB2 UDB database, determining size requirements (procedure) 88, 274
- time zones
 - global, about 202, 331
- triggers
 - upgrading, about customized triggers (development environment) 66, 253
- troubleshooting
 - Siebel repository (development environment) 144
 - Siebel Repository, upgrading (development environment) 121, 168, 306, 316
- U**
- UI objects
 - copying, guidelines for 135
- UNIX
 - DB2 8-KB tablespaces and buffers 173, 321
 - DB2 UDB, creating 32-KB tablespace and buffers 87, 273
 - DB2 UDB, creating additional tablespaces and buffers 85, 271
- upgphys.ksh
 - errors.rtf 295
- Upgrade Ancestor. *See* upgrading inheritance
- Upgrade Planning Worksheet
 - development environment, completing before starting upgrade 59
- UPGRADE_DB 294
- upgrading inheritance
 - about 127, 128
 - copy of object, creating 131
 - descendent or copies of an object, viewing 134
 - enhancements, process of applying 128
 - parent to descendents, propagating changes from 136
 - repositories, about comparison of 130
- UI objects and business components, guidelines for copying 135
- upgrade ancestor for an object, choosing 129
- using scenario 128

- upgrading, planning for
 - before starting, steps 25
 - database server configuration, about
 - verifying and database growth (table) 68, 254
 - MS SQL Server database, about 71, 256
 - Oracle database, steps before
 - upgrading 73, 259
 - process (diagram) 18
 - resources 26

V

- visibility rules
 - repository upgrade, about (development environment) 97, 282

W

- Windows
 - DB2 8-KB tablespaces and buffers,
 - dropping 173, 321
 - DB2 UDB, creating 32-KB tablespace and buffers 87, 273
 - DB2 UDB, creating additional tablespaces and buffers 85, 271
 - servers, upgrading (development environment) 107, 290
 - Siebel Database Server software,
 - configuring for upgrade (development environment) 110, 297
 - Siebel eBusiness Application, uninstall failure 105

